AD NUMBER

AD894895

NEW LIMITATION CHANGE

TO

Approved for public release, distribution unlimited

FROM

Distribution authorized to U.S. Gov't. agencies only; Test and Evaluation; APR 1972. Other requests shall be referred to Air Force Flight Dynamics Lab., Wright-Patterson AFB, OH 45433.

AUTHORITY

AFWAL ltr, 13 Mar 1981

*AD 894 895

AFFDL-TR-71-5
PART II, VOLUME II

SUBSONIC UNSTEADY AERODYNAMICS FOR GENERAL CONFIGURATIONS

PART II

VOLUME II COMPUTER PROGRAM N5KA

J. P. GIESING
T. P. KALMAN
W. P. RODDEN

TECHNICAL REPORT AFFDL-TR-71-5, PART II, VOLUME II

APRIL 1972

Approved for public release; distribution is unlimited.

~~Distribution limited to U.S. Government agencies only; test and evaluation; statement applied in September 1971. Other requests for this document must be referred to AF Flight Dynamics Laboratory, (FY), Wright-Patterson AFB, Ohio 45433.~~

AIR FORCE FLIGHT DYNAMICS LABORATORY
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

AIR FORCE: 15-5-72/50

SUBSONIC UNSTEADY AERODYNAMICS FOR GENERAL CONFIGURATIONS

PART II

VOLUME II COMPUTER PROGRAM N5KA

J. P. GIESING
T. P. KALMAN
W. P. RODDEN

Approved for public release; distribution is unlimited.

~~Distribution limited to U. S. Government agencies only; test and evaluation; statement applied in September 1971. Other requests for this document must be referred to AF Flight Dynamics Laboratory (FY), Wright-Patterson AFB, Ohio 45433.~~

CODE
23

FOREWORD

This report was prepared by the Douglas Aircraft Company, Aircraft Division, Long Beach, California, for the Aerospace Dynamics Branch, Vehicle Dynamics Division, Air Force Flight Dynamics Laboratory, Wright-Patterson Air Force Base, Ohio under Contract F33615-70-C-1167. This research was conducted under Project 1370, "Dynamic Problems in Military Flight Vehicles", and Task 137003, "Prevention of Dynamic Aeroelastic Instabilities in Advanced Military Aircraft." Mr. S. J. Pollock of the Aerospace Dynamics Branch was Task Engineer.

This report consists of two parts with two volumes for each part. This volume, Volume II of Part II is the Computer Program N5KA. Volume I of Part II contains a method which uses an image system and an axial singularity system to account for the effects of the bodies. Volume I of Part I contains the method of direct application of nonplanar lifting surface elements, and Volume II of Part I is the Computer Program H7WC.

The work reported herein was conducted during the period of December 1969 to August 1971.

The Principal Investigator was Joseph P. Giesing. Mrs. T. P. Kalman was responsible for the computer programming and Dr. W. P. Rodden was a McDonnell Douglas Company Consultant. Others have made significant contributions to this project including Messrs. D. H. Larson, D. S. Warren, and W. E. Henry.

The contractor's designation of this report is MDC-J0944. The report was released by the authors in August 1971 for publication as an AFFDL Technical Report.

This technical report has been reviewed and is approved.

Walter J. Mykytow
WALTER J. MYKYTOW
Asst. for Research & Technology
Vehicle Dynamics Division

ABSTRACT

A technique for predicting steady and oscillatory aerodynamic loads on general configurations has been developed which is based on the Doublet-Lattice Method and the method of images. Chord- and spanwise loading on lifting surfaces and longitudinal body load distributions are determined. Configurations may be composed of an assemblage of bodies (elliptic cross sections and a distribution of width or radius) and lifting surfaces (arbitrary planform and dihedral, with or without control surfaces). Loadings predicted by this method are required for flutter, gust, frequency response and static aeroelastic analyses and may be used to determine static and dynamic stability derivatives. Volume I presents the theory and calculated results while Volume II presents the details of the computer program used to implement the theory.

CONTENTS

Page

NOMENCLATURE

A	Reference total area
a	Average body width
a_o	Local body width
$\bar{a}$	Radius of curvature
b	Average body height
b_o	Local body height
$\bar{C}$	Position vector to center of curvature
C_ℓ	Rolling moment coefficient (moment/qA_s) (+ right wing down)
C_m	Pitching moment coefficient (moment/$qA\bar{c}$) (+ nose up)
C_n	Yawing moment coefficient (moment/qA_s) (+ nose right)
C_p	Pressure coefficient
C_Y	Side force coefficient (Force/qA) (+ out right wing)
C_Z	Vertical force coefficient (Force/qA) (+ up)
c	Local chord length
$\bar{c}$	Reference chord length
c_m	Local pitching moment coefficient
c_n	Local normal force coefficient
c.p.	Center of pressure
D	Matrix relating normalwash to lifting pressures for lifting surface elements
D_I	Matrix relating normalwash to lifting pressures for image elements
$\bar{\bar{D}}$	Matrix relating normalwash to lifting pressures for elements and all their images
$\bar{D}$	Matrix relating normalwash to lifting pressures for elements plus their images plus the contributions due to symmetry and ground effect

D_T	Partitioned matrix $[\bar{D} \mid \bar{E}]$, relating normalwash to lifting pressures and doublet strengths
D_θ	Matrix relating the flow normal to a body surface (at the meridian angle θ) to the lifting pressure for elements and their images
$D^{(y)}, D^{(z)}$	Matrix relating the average side- or upwash at a body due to lifting surface elements
D2D	Matrix relating the doublet strength to the local up- or side-wash using quasi-steady, two-dimensional slender-body theory
d	Spacing of doublets or vortices within slender bodies (simulation of body aspect ration (b/a))
E	Matrix relating normalwash to axial doublet strengths
$\bar{E}$	Matrix relating normalwash to axial doublet strengths with the effects of symmetry and ground effect included
$E^{(y)}, E^{(z)}$	Matrix relating the normalwash to y- or z-oriented axial doublets
e	Lifting surface element semi-width; also cross-sectional element semi-width
$F^{(z)}, F^{(y)}$ $F_y^{(z)}, F_y^{(y)}$	Total force on a body due to a point pressure doublet. Subscript indicates direction of force; superscript indicates direction of pressure doublet
f	Nondimensional deflection. Also function involving Hankel functions
$H_\nu^{(2)}$	Hankel function of the second kind of order ν
h	Deflections normal to a lifting surface
h_y, h_z	Deflections of a body in y- and z-directions, respectively
$\vec{i}, \vec{j}, \vec{k}$	Unit vectors in x-, y- and z-directions, respectively
$\vec{i}_F$	Unit vector in the direction of the body force
K	Velocity kernel function; the normalwash due to a point pressure doublet; also $(a_0^2 - b_0^2)/4$
K_ϕ	Potential kernel function; the potential due to a point pressure doublet
k_r	Reduced frequency $(\omega\bar{c}/2U_\infty)$
$\bar{k}$	$\omega rM/U_\infty$

L	The normalwash due to a potential doublet
L_ϕ	The potential due to a potential doublet
M	Mach number; also normalwash due to a point source; also moment
$\vec{N}$	Orientation of pressure doublet
$\vec{n},\vec{t}$	Outward normal and tangent vectors
p	Function involving Hankel functions
Q	Generalized force; also modified acceleration potential
q	Dynamic pressure
$\bar{q}$	Generalized modal coordinate
R	$\sqrt{(x - \xi)^2 + \beta^2 r^2}$
r	$\sqrt{(y - \eta)^2 + (z - \zeta)^2}$
$\tilde{r}$	$(a + b)/2$
s	semi-span
U_∞	Freestream velocity
w	Normalwash boundary values
w_i	Normalwash due to image lifting surface elements
w_n	Normalwash due to body interference doublet distribution
w_R	$w_s + w_I$
w_s	Normalwash due to lifting surface elements
w_T	$w - \Delta w$
$\bar{w}$	Normalwash in the circle plane
x,y,z	Coordinates of a receiving point
XM	Coordinate about which moments are taken
α	angle of attack
β	$\sqrt{1 - M^2}$
γ	Dihedral angle: γ_r, receiving point, γ_s, sending point
Γ	Vortex strength

ΔC_p	Lifting pressure
$\Delta\bar{Q}$	Modified acceleration potential jump
Δw	Normalwash due to slender body elements
Δx	Longitudinal length of lifting surface box
$\Delta\xi$	Longitudinal length of axial element
$\Delta\phi$	Potential jump
δ	Symmetry plane indication (1 symmetry, 0 no symmetry, -1 antisymmetry); also a delta function; also a virtual displacement
δA	Elemental area
ε	Ground-effect indication (-1 ground effect, 0 no ground effect, 1 antiground effect)
ζ	z-coordinate of sending point
η	y-coordinate of sending point
$\bar{\eta}$	Lateral coordinates in the plane of the lifting surface
θ	Meridian angle for a body of circular cross section
λ	Sweep of 1/4-chord of lifting surface element; also inclination angle in z-y-plane of a cross-sectional surface element
μ_d	Quadrupole strength
μ_n	Doublet strength of interference-body elements
μ_s	Doublet strength of slender-body elements
$\bar{\mu}_\nu$	Multipole strength in circle plane; ν gives order of pole
$\tilde{\mu}_y, \tilde{\mu}_z$	Doublet strength of modified acceleration potential distribution in y- and z-directions; also reduction factors for image doublets
ξ	x-coordinate of sending point
$\bar{\rho}$	Distance from center curvature to external singularity
σ	Source strength
ϕ	Velocity potential
Ω	Acceleration potential
ω	Frequency

ξ_c	Center of axial-body element
ξ_1	Leading edge of body element
ξ_2	Trailing edge of body element

Subscripts and Superscripts

a	Body axis
b	Body
I	Image
LL	Lower left-hand quadrant
LR	Lower right-hand quadrant
n	Residual or interference flow
r,s	Receiving and sending points, respectively
UL	Upper left-hand quadrant
UR	Upper right-hand quadrant
s	Steady
y,q	y- and z-directions
θ	On the body surface
1,2	Planar and nonplanar parts, respectively
1/4	Quarter chord of element

1.0 INTRODUCTION

Program N5KA is the result of implementing the equations of Section 2.0, Part II, Vol. I (Reference 1) for the computer. The organization of these computations is outlined as follows:

1) All of the data required (except modal data) are generated from the input data in program Segment No. 2.

2) The influence coefficient matrix $[D_T]$ is generated in program Segment No. 3. This matrix relates the normalwash, upwash and sidewash to the lifting surface pressures and body axial doublet strengths in the z- and y-directions.

3) The normalwash, sidewash and upwash flow fields, Δw, caused by the slender body elements is generated in program Segment No. 5 for all modes. Modal data is read in and organized in program Segment No. 4. Currently the source distributions for steady flow are not included.

4) The final normalwash (also side and upwash) boundary condition, w_T, caused by the motions of the lifting surfaces and the slender body flow field, Δw, is generated in program Segment No. 6.

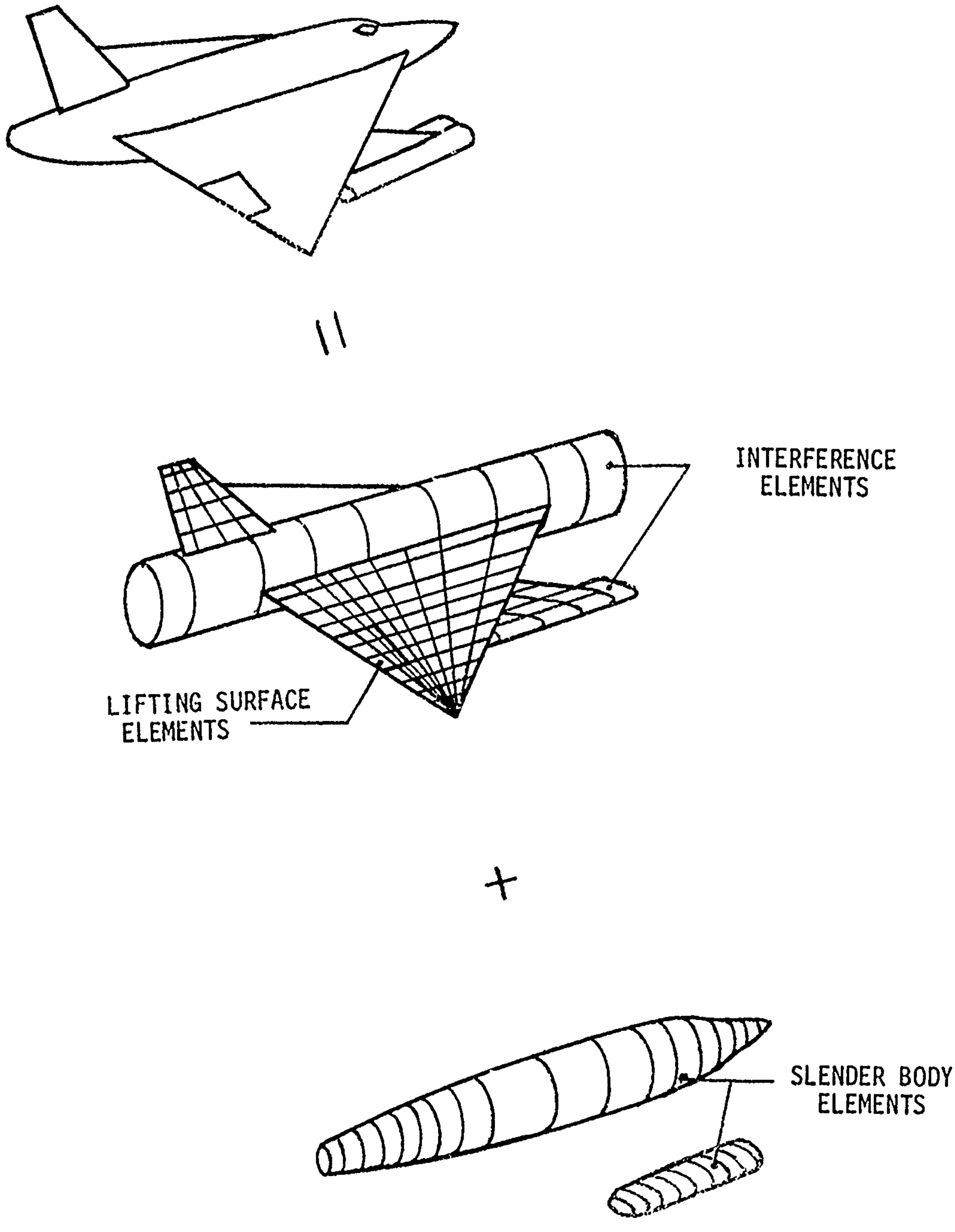
5) The augmented matrix $[D_T \mid w_T]$ is formed in program Segment 1 and solved in program Segment 7 for the lifting surface pressures and body doublet strengths (both in z- and y-directions).

6) The solution obtained in program Segment 7 is used to calculate the body forces and moments in either program Segment 8 or program Segment 9 depending on the method of calculations desired.

7) The lifting surface pressures and body axial force and moment distributions are integrated to form aerodynamic coefficients in program Segment No. 10. Also in this segment the body forces are redistributed.

8) The lifting surface pressures and body axial force and moment distributions are integrated to form the generalized forces in program Segment 11.

Configurations are built up of lifting surface panels and bodies to any degree of complexity desired (see Sketch 1.0-1). A lifting surface panel is a



INTERFERENCE ELEMENTS
LIFTING SURFACE ELEMENTS
SLENDER BODY ELEMENTS

SKETCH 1.0-1

trapezoid (two edges parallel to the x-axis) defined by the coordinates of its corner points. The panel is divided arbitrarily chordwise and spanwise to produce a surface of elements or boxes. Hinge lines, fold lines, and lines of intersection (of two or more panels) must lie on box boundaries. Lifting surface panels that lie one in back of another must be aligned so that those box edges lying in the same plane coincide. Part I, Volume I of this report[2] provides additional information on the distribution of lifting surface boxes (pages 49-50). Bodies are defined in two ways: 1) a tube of constant cross section, of aspect ratio b/a, divided into interference elements and 2) an equivalent body of elliptic cross section of varying radius divided up into slender body elements. The interference elements are provided to complete the body interference and are concentrated in regions of maximum interference while the slender body elements and radius distribution are used to obtain the slender body flow field of the body. It is important that the radius distribution be accurate because a numerical derivative of the radius must be taken with respect to x. Also provided, for bodies, are one or two distributions of pickup points. Pickup points are the points on the body surface where the normalwash is to be determined in order to calculate an average up- or side wash. The distribution of pickup points is given in terms of θ, which for a circle is the meridian angle. For an ellipse θ is defined as $\tan^{-1}\left(\frac{az}{by}\right)$. There are two possible θ distributions; one for regions where many (e.g. 8) points are required and regions where few (e.g., 4) are required.

When the actual lifting-surface/body geometry is built up the constant section tube is used to represent the body. It is important that the lifting surfaces adjacent to bodies be attached to the body surfaces without any gaps. Even small gaps cause a reduction in load near the gap. The relationship between lifting surface panels and bodies may be incorrect because the average tube is used instead of the actual body shape. In order to correct this difficulty and obtain the proper flow field at the lifting surface panel due to the slender body elements, a shifting of the panel is provided for. The shift of a panel may be different for different bodies. Thus the quantities $\Delta\eta_b^{(p)}$, $\Delta\zeta_b^{(p)}$ are input into the program, where _p_ indicates panel and _b_ indicates body.

To increase the efficiency of the method a further input is required for each panel. This input identifies the bodies "associated" with that panel.

Associated bodies are those that are required to possess an image of the panel in question. It is important to have only as many images as is necessary for accuracy since each image doubles the number of kernel functions that are to be calculated. It may even be desirable to break up lifting surfaces into more panels than necessary so that panels, located a considerable distance from the body, may not be required to possess an image within that body.

Symmetry planes and ground effect are included by the use of the input quantities δ and ε. Symmetry, no symmetry and antisymmetry are activated by setting δ equal to 1, 0 and -1 respectively. The plane of symmetry is the $y = 0$ plane. Ground effect, no ground effect and antiground effect are activated by setting ε equal to -1, 0 and 1 respectively. The ground effect plane is the $z = 0$ plane. Thus if ground effect is desired the configuration must be placed a distance above the $z = 0$ plane.

Polynomial modes are input for panels (motion normal to the surface) and for bodies (motions in the z- and y-directions). The coefficients of the polynomials are the input quantities to the program.

The following list gives the program limits:

1) The maximum number of unknowns, i.e. the total number of all the lifting surface elements plus the interference body elements, is 500.

2) The maximum number of modes is either 50 or $\frac{\text{NWORK}}{\text{NTOT}}$ whichever is smaller, where NWORK = 10000 and NTOT is the total number of unknowns for the case.

3) The maximum number of panels is 99 while the maximum number of bodies is 10.

4) The maximum number of spanwise strips per panel is 50 while the maximum number of chordwise boxes per strip is 50.

5) The maximum number of body interference elements (for all bodies) is 100 while the maximum number of slender body elements is 200. The work area dimension, NWORK, imposes the following overall restriction: 4 (NSTRIP + NBZ + NBY + NTZS + NTYS) must be $\leq$ NWORK, where NSTRIP is the total number of strips, NBZ is the number of z-oriented bodies, NBY is the number of y-oriented bodies, and NTZS, NTYS are the total number of slender body elements with z- and y-orientations respectively.

6) The maximum number of modal coefficients for panels is 150. The maximum number of modal coefficients for bodies is 150 for z-motions and 150 for body y-motions.

7) The maximum number of reduced frequencies is 6.

It is important that none of the above maximum values is exceeded. Not all maximums can be utilized at the same time without violating others. For example if the maximum number of spanwise strips and chordwise boxes is input then the total number of lifting surface boxes will exceed the maximum number of unknowns. The maximums outlined above are tailored to allow the computer program to fit into a core (360/65) of 260 K bytes. If more core is available, the user may wish to increase the dimension (NWORK) of the work array (WORK, see Section 5.5.1) in order to accommodate larger cases in the program.

2.0 INPUT PROCEDURE AND EXAMPLE CASE

2.1 Input Sheets

The input sheets for program N5KA are shown on the next two pages. The first three cards represent general data that is input once per case. The next four* cards (sequence numbers 4, 5, 6 and 7) represent panel data that is repeated per panel.

If the data includes bodies also, the next input card (#8) contains general information for the first body of the case. The subsequent two** cards (sequence numbers 9 and 10) are interference body element data, and the next two** cards (#11 and #12) represent slender body element data. Next, the θ_1- and θ_2-arrays are input (cards #13 and 14) which describe the angular distribution of points on the surface of interference body elements. Card 15 identifies the sections of the interference body for which θ_1-distribution is specified. The next card (#16) contains the y- and z-shift information for all panels if this is also desired (see control items in card 8).

Cards 8 through 16 represent all the body data required and are repeated per body. Note, that bodies oscillating in the z-direction are input first, then bodies that can oscillate both in the z- and y-directions, and finally, bodies that oscillate in the y-direction only.

The last four† cards represent the polynomial mode information for the case. Card 17 contains the modal data for z-oriented bodies, card 18 contains the same for y-oriented bodies, and card 19 represents modal data for panels. Modal data may be input in any order with a maximum of three sets of data per card, a total of 100 sets of data per case, and only the nonzero modal coefficients need to be input. Card 20 terminates the reading of modal data; always input card 20 as the last card following all modal information. If the case specifies more than one reduced frequency, all modal data cards are to be repeated for each reduced frequency (see card #3).

A detailed description of all data items is given following the input sheets for program N5KA.

---

* There may be more than four cards in order to present all θ and τ values.

** There may be more than two cards in order to present all ξI, RI elements for interference bodies, and ξS, RS elements for slender bodies.

† There may be more than four cards needed to enter all modal information.

DATE ____________

PROGRAM NO. N 5 K A (columns 73 74 75 76)

1–10 (±)	11–20 (±)	21–30 (±)	31–40 (±)	41–50 (±)	51–60 (±)	61–70 (±)	SEQ. NO. (77–80)
		HEADER				I B F S	01
M	A	S	C	XM	δ ε NP NB NK	MK1 MK2 N1 N2 N3 N4	02
		k_{1}					03
X_1	X_2	X_3	X_4	Y_1	Y_2		04
Z_1	Z_2	NC	NS	ASSOCIATED BODIES			05
		θ					06
		τ					07
z_c	y_c	a	AR	NBE NSBE NZY NRI NRS	NSH NT1 NT2		08
		εI					09
		RI					10

DATE ______________

PROGRAM NO. 73 74 75 76: N 5 K A

Cols 1–10	Cols 11–20	Cols 21–30	Cols 31–40	Cols 41–50	Cols 51–60	Cols 61–70	SEQ. NO. (77–80)
		ϵS					11
		BS					12
		$\theta 1$					13
		$\theta 2$					14
FIRST	LAST	. . .	ELEMENTS	FØR $\theta 1$			15
PANEL NØ.	$\Delta\eta$	$\Delta\zeta$	PANEL NØ.	$\Delta\eta$	$\Delta\zeta$		16
A, Z NO NRZ N	$az_{qn}^{(b)}$	NO NRZ N	$az_{qn}^{(b)}$	. . . modal data	for z-bodies		17
A, Y NO NRY N	$ay_{qn}^{(b)}$	NO NRY N	$ay_{qn}^{(b)}$	. . . modal data	for y-bodies		18
A NO NRP M N NB	$a_{qmn}^{(p)}$	NO NRP M N NB	$a_{qmn}^{(p)}$	. . . modal data	for panels		19
-1							20

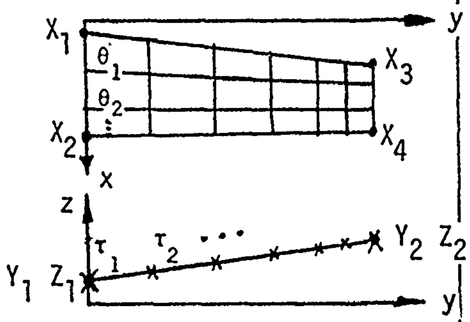
2.2 Description of Input Data

CARD NO.	ITEM NO.	MNEMONIC	SYMBOL	CARD COLUMN	FIELD	SOURCE	DESCRIPTION
	0	HDR(15)		1-60	15A4		Header information
1	1	IBFS		61-70	I10		Body force calculation method flag, alternate #1 IBFS = 1;* alternate #2 IBFS = 0.
	2	FMACH	M	1-10			Mach number, usual definition
	3	REFA	A	11-20			Reference area; usually total area of both wings
	4	REFS	S	21-30	5F10.0		Reference semispan
	5	REFC	$\bar{c}$	31-40			Reference chord; usually average chord of wing
2	6	XM	XM	41-50		MAIN	Moment axis
	7	ND	δ	51-52	I2		Symmetry flag (y = 0 plane) δ = 1 for symmetry δ = -1 for antisymmetry δ = 0 for no symmetry
	8	NE	ε	53-54	I2		Second symmetry flag (z = 0 plane) ε = 1 for biplane effect (symmetry)

*Use #1 for circular and #2 for elliptic cross-sections

CARD NO.	ITEM NO.	MNEMONIC	SYMBOL	CARD COLUMN	FIELD	SOURCE	DESCRIPTION
							ε = -1 for ground effect (antisymmetry) ε = 0 no symmetry
	9	NP, NOPAN		55-56	I2		Total number of panels on all lifting surfaces
	10	NB		57-58	I2		Total number of bodies
	11	NK		59-60	I2		Total number of reduced frequencies; max. 6 per case
	12	MK1		61-63	I3		Sequence number of first box on first panel representing a body surface, whenever this body is at zero incidence; otherwise MK1 = 0
	13	MK2		64-66	I3		Sequence number of last box on last panel representing a body surface, whenever this body is at zero incidence; otherwise MK2 = 0. Note that panels on body surfaces need not be input last
2	14	N1, NPR1		67	I1	MAIN	Print flag for solutions; N1 = 1 means all solutions are printed, N1 = 0 means no print. Usual setting is N1 = 0

CARD NO.	ITEM NO.	MNEMONIC	SYMBOL	CARD COLUMN	FIELD	SOURCE	DESCRIPTION
	15	N2, NPR2		68	I1		Control flag for pressures and generalized forces. N2 = 0 means no pressures and no generalized forces; N2 = 1 means pressures printed, and generalized forces computed according to AGARD definition; N2 = 2 means pressures and conventional generalized forces - see Sec. 5.11.1
	16	N3		69	I1		Data flag; N3 = 1 means DT matrix print; N3 = 0 means no print. Usually N3 = 0
2	17	N4		70	I1	MAIN	Detail print flag for subroutines RDMØDE, SB, WANDWT and BFM; N4 = 0 means no print; N4 = 1 means detail print in subroutine BFM only; N4 = 2 means detail print in all four subroutines. Usually N4 = 0
3	18	FREQ(10)	k_r	1-60	6F10.0		Array of reduced frequencies $k_r = \frac{\omega\bar{c}}{2U_\infty}$

CARD NO.	ITEM NO.	MNEMONIC	SYMBOL	CARD COLUMN	FIELD	SOURCE	DESCRIPTION
	19	X1	X_1				Panel edge coordinates
	20	X2	X_2				(X_1,Y_1,Z_1) - inboard leading edge
4	21	X3	X_3	1-60	6F10.0		(X_2,Y_1,Z_1) - inboard trailing edge
	22	X4	X_4				(X_3,Y_2,Z_2) - outboard leading edge
	23	Y1	Y_1				(X_4,Y_2,Z_2) - outboard trailing edge
	24	Y2	Y_2				
5	25	Z1	Z_1	1-20	2F10.0		
	26	Z2	Z_2				
	27	NC	nc	21-30	2I10	DATA	Number of chordwise divisions for panel
	28	NS	ns	31-40			Number of spanwise divisions for panel
	29	NAB(10)		41-60	10I2		Associated bodies; a max. of six per panel
6	30	TH(50)	θ_i	1-60	6F10.0		Fractional chordwise divisions for panel. Usually varies from 0 at leading edge to 1.0 at trailing edge
7	31	TAU(50)	τ_j	1-60	6F10.0		Fractional spanwise divisions for panel. Usually varies from 0 at inboard edge to 1.0 at outboard edge

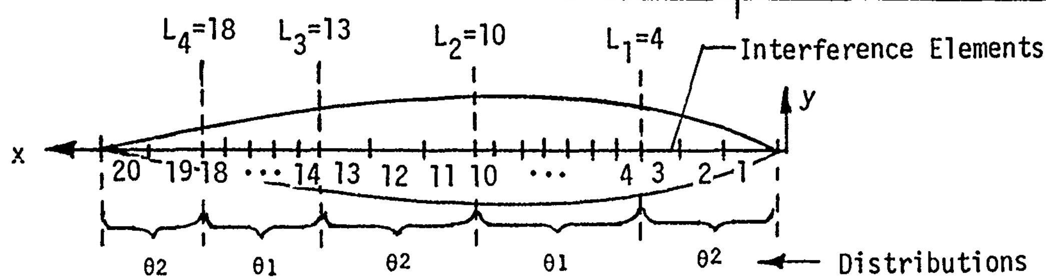
Repeat Items #19 through 31 for all panels

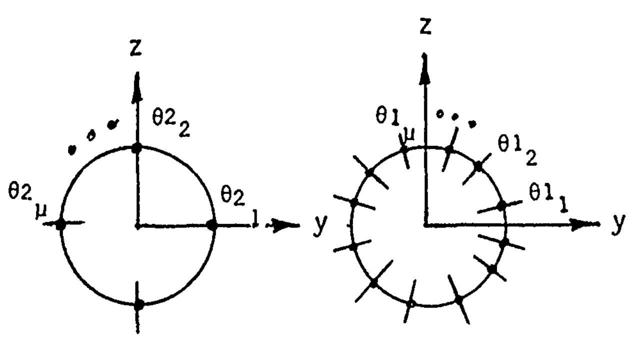
CARD	ITEM	MNEMONIC	SYMBOL	CARD COLUMN	FIELD	SOURCE	DESCRIPTION
8	32	ZC	z_c	1-10	4F10.0	DATA	z-coordinate of body axis
	33	YC	y_c	11-20			y-coordinate of body axis
	34	RAD	a	21-30			Average characteristic semi-width of body
	35	AR	AR	31-40			Cross-sectional aspect ratio of body
	36	NBE		41-43	2I3		Number of interference body elements
	37	NSBE		44-46			Number of slender body elements
	38	NZY		47-48	I2		NZY=1 - z-oriented body NZY=2 - z-and y-oriented NZY=3 - y-oriented body Input bodies in this order, i.e. z-oriented bodies first, then z- and y-, then y- bodies
	39	NRI		49-50	I2		Interference 'radius' flag; NRI = 1 - RI - array is input (see below); NRI = 0 - RI-array is not input, but rather is taken as 'a', i.e. RI_i=a for all i = 1, NBE
	40	NRS		51-52	I2		Slender body 'radius' flag; NRS=1 - RS-array is input (see below) NRS=0 - RS-array is not input; instead, RS_i=a, all i = 1, NSBE

CARD NO.	ITEM NO.	MNEMONIC	SYMBOL	CARD COLUMN	FIELD	SOURCE	DESCRIPTION
	41	NSH		53-54	I2	DATA	Number of $\Delta\eta$-$\Delta\zeta$ pairs for body - see Items #51 - 53
8	42	NT1	$N\theta_1$	55-56	I2		Number of elements in the $\theta 1_\mu$-array - see item #48
	43	NT2	$N\theta_2$	57-58	I2		Number of elements in the $\theta 2_\mu$-array - see item #49. Note that, if NT2=0, the $\theta 2_\mu$-array is not input
9	44	XII(100)	ξI_i	1-60	6F10.0		x-coordinates of inter-ference* body element endpoint; i=1, (NBE+1)
10	45	RI(100)	RI_i	1-60	6F10.0		Average char-acteristic semi-widths of inter-ference* body element endpoint; i=1, (NBE+1) Omit RI if NRI = 0
11	46	XIS(100)	ξS_i	1-60	6F10.0		x-coordinates of Slender body element endpoints; i+1, (NSBE+1)
12	47	RS(100)	RS_i	1-60	6F10.0		Average char-acteristic semi-widths of Slender body element endpoints; i+1, (NSBE+1) Omit RS if NRS = 0

* Omit Items #44 and 45 if NBE = 0

CARD NO.	ITEM NO.	MNEMONIC	SYMBOL	CARD COLUMN	FIELD	SOURCE	DESCRIPTION
13	48	TH1(24)	$\theta1_{\mu}$	1-60	6F10.0		Angular orientation of the points 'μ' on interference body surfaces - first array, μ = 1, NT1
14	49	TH2(24)	$\theta2_{\mu}$	1-60	6F10.0	DATA	Second array of θ_{μ}'s for interference bodies; μ = 1, NT2 Omit this item if NT2 = 0
15	50	L1 L2 L3 L4 L5 L6		1-60	6I10		First, and Last elements for interference body with $\theta1_{\mu}$-distribution; a max. of three pairs per body



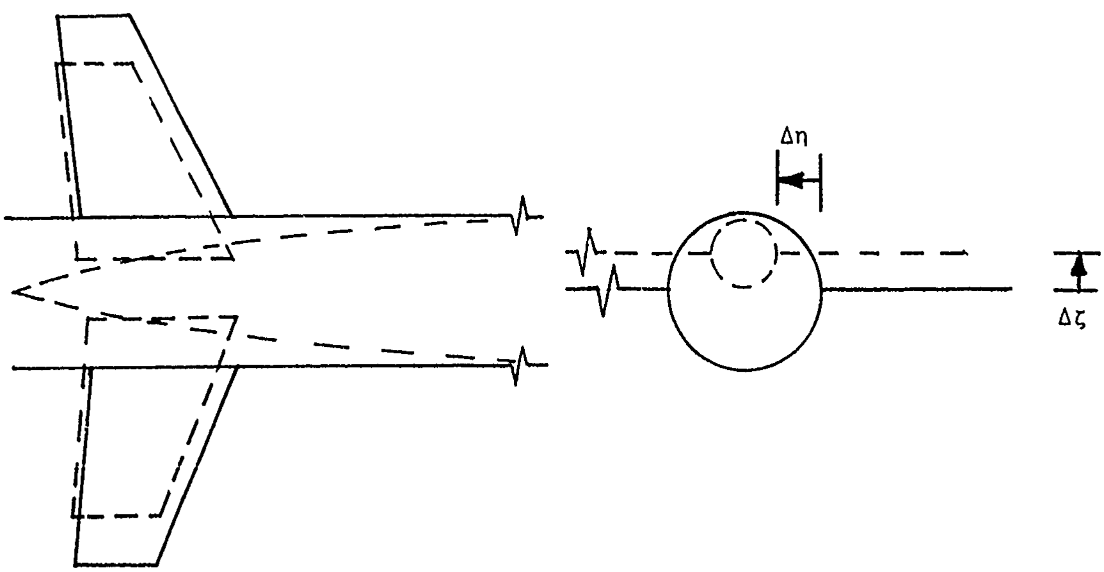


θ2 Distribution θ1 Distribution

Items #48 and #49 are input in degrees

Omit Items #48-50 if NBE = 0

CARD NO.	ITEM NO.	MNEMONIC	SYMBOL	CARD COLUMN	FIELD	SOURCE	DESCRIPTION
	51	NCD1		1-10	I10		Panel member for $\Delta\eta$, $\Delta\zeta$ pair
16	52	CD2	$\Delta\eta_i$	11-20	2F10.0		y-shift of panel, first set
	53	CD3	$\Delta\zeta_i$	21-30		DATA	z shift
		NCD4		31-40	I10		
		CD5	$\Delta\eta_{i+1}$	41-50	2F10.0		Another set of the above three items.
		CD6	$\Delta\zeta_{i+1}$	51-60			Repeat for all i = 1, NSH



Omit Items #51-53 if NSH = 0

Repeat Items #32 through 53 for all bodies; omit same if there are no bodies for case.

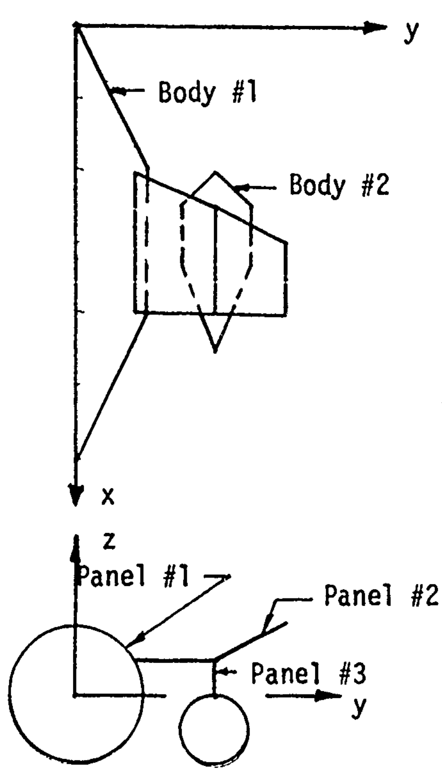
CARD NO.	ITEM NO.	MNEMONIC	SYMBOL	CARD COLUMN	FIELD	SOURCE	DESCRIPTION
17	54	AZ	AZ	1-2	I2	RDMODE	Punch 'AZ' in cc 1-2 of all modal data cards for z-oriented bodies
	55	q	NQ	3-4 23-24 43-44	I2		Mode number for the first / second / third set of modal data
	56	r	NRZ	5-7 25-27 45-47	I3		Body number first / second / third value
	57	n	N	8 28 48	I1		Power of $x/\bar{c}$ in the mode polynomial first / second / third value
	58	$az_{qn}^{(b)}$		11-20 31-40 51-60	F10.0		Coefficient of $(x/\bar{c})^n$ in the mode polynomial first / second / third value
18	59	AY	AY	1-2	I2		Punch 'AY' in cc 1-2 of all modal data cards for y-oriented bodies
	60 61 62 63	q r n $ay_{qn}^{(b)}$					As Items #55-58, but now for y-oriented bodies
	64	A	A	1	I1		Omit cards #17-18 if NB=0 Omit card #17 if NBZ=0 Omit card #18 if NBY=0 Punch 'A' in cc 1 of all modal data cards for panels

CARD NO.	ITEM NO.	MNEMONIC	SYMBOL	CARD COLUMN	FIELD	SOURCE	DESCRIPTION
19	65	NQ	q	3-4 23-24 43-44	I2	RDMODE	Mode number for the first / second / third set of modal data
	66	NRP	p	5-7 25-27 45-47	I2		Panel number - 3 sets
	67	M	m	8 28 48	I1		Power of $\bar{y}/\bar{c}$ in the mode polynomial, where $\bar{y}$ is a 'spanwise' coordinate along the lifting surface, e.g., for fins $\bar{y} = z$
	68	N	n	9 29 49	I1		Power of $x/\bar{c}$ in the mode polynomial
	69	N8	N8	10 30 50	I1		Flag that sets the origin of the spanwise coordinate $\bar{y}$. N8=0 means origin of coordinates; N8=1 means inboard edge of panel. See Sec. 5.4.1 (Subroutine RDMODE)for details.
	70		$a_{qmn}^{(p)}$	11-20 31-40 51-60	F10.0		Coefficient of $(x/\bar{c})^n(\bar{y}/\bar{c})^m$ in the mode polynomial

CARD NO.	ITEM NO.	MNEMONIC	SYMBOL	CARD COLUMN	FIELD	SOURCE	DESCRIPTION
20	71	-1	-1	1-2	I2	RDMODE	Punch '-1' in cc 1-2 of card following last modal data card for case
Note that only the nonzero coefficients need to be input; each of the cards #17, 18 and 19 accommodate 3 sets of modal data. If NK>1, the modal data cards #17-20 have to be repeated for each reduced frequency, i.e., input these NK times.							

2.3 Example Case Input Sheets

The following case is to be viewed only as an example of the proper input procedure and not as an optimum idealization. The configuration is shown in Sketch 2.3-1.



SKETCH 2.3-1

k_r = 0.5
M = 0.85
A = 6.4
S = 3.0
$\bar{c}$ = 1.5
XM = 2.0
δ = 1
ε = 0

The configuration consists of 3 panels and 2 bodies. The panels are given as follows:

(1) $X_1 = 2.0$ $X_2 = 4.0$ $X_3 = 2.5$ $X_4 = 4.0$

$Y_1 = 0.86603$ $Y_2 = 2.0$ $Z_1 = 0.5$ $Z_2 = 0.5$

$\theta = 0.0, 0.5, 1.0$ and $\tau = 0.0, 0.5, 1.0$

Associated body is body #1.

(2) $X_1 = 2.5$ $X_2 = 4.0$ $X_3 = 3.0$ $X_4 = 4.0$

$Y_1 = 2.0$ $Y_2 = 3.0$ $Z_1 = .05$ $Z_2 = 1.0$

$\theta = 0.0, 0.5, 1.0$ and $\tau = 0.0, 0.5, 1.0$

Panel #2 has no associated bodies.

(3) $X_1 = 2.5$ $X_2 = 4.0$ $X_3 = 2.5$ $X_4 = 4.0$

$Y_1 = 2.0$ $Y_2 = 2.0$ $Z_1 = 0.5$ $Z_2 = 0.0$

$\theta = 0.0, 0.5, 1.0$ and $\tau = 0.0, 1.0$

Associated body is body #2.

The body data is as follows:

Body #1 is z-oriented with $Z_C = 0.0$, $Y_C = 0.0$ and is divided into 3 interference elements and 5 slender body elements with end points

$$\xi I = 0.0, \quad 2.0, \quad 4.0, \quad 6.0$$

$$\xi S = 0.0, \quad 1.0, \quad 2.0, \quad 4.0, \quad 5.0, \quad 6.0$$

The interference body radius is $a = 1.0$, and the array of the slender body element radii is

$$RS = 0.0, \quad 0.5, \quad 1.0, \quad 1.0, \quad 0.5, \quad 0.0.$$

The second element of the interference body has θ_1 distribution on its surface, while the other elements have θ_2 distribution:

$$\theta_1 = 0, \quad 60, \quad 120, \quad 180, \quad 240, \quad 300$$

$$\theta_2 = 0, \quad 90, \quad 180, \quad 270.$$

Body #2 is both z- and y-oriented with $Z_c = -0.5$ and $Y_c = 2.0$. The element endpoints are defined by

$$\xi I = 2.0, \quad 2.5, \quad 3.25, \quad 4.0, \quad 4.5$$

$$\xi S = 2.0, \quad 2.25, \quad 2.5, \quad 3.25, \quad 4.0$$

The interference body radius is 0.5 and the slender body radii are given by the array

$$RS = 0.0, \quad 0.25, \quad 0.5, \quad 0.0$$

All interference elements of body #2 have θ_1 distribution:

$$\theta_1 = 45, \quad 135, \quad 225, \quad 315.$$

The configuration is given three modes of motion:

Mode #1 is a plunging motion. The coefficients of motion are given by $a_{00}^{(p)} = \cos\gamma$ for all panels p, where γ = dihedral angle of panel, and $az_0^{(b)} = 1.0$ for bodies with z-orientation.

Mode #2 is a pitching motion. The coefficients are $a_{01}^{(p)} = \cos\gamma$ for all panels p, and $az_1^{(b)} = 1.0$ for the z-bodies.

Mode #3 is a rolling motion. The coefficients are $a_{00}^{(p)} = (Y_1 \cos\gamma + Z_1 \sin\gamma)/\bar{c}$ and $a_{10}^{(p)} = 1.0$ for all panels p. Body #1 has no motion in this mode while body #2 has two modal coefficients $az_0^{(2)} = 2.0/\bar{c}$ and $ay_0^{(2)} = 0.5/\bar{c}$.

The input sheets are shown on the next several pages, while the output for the case is found on the subsequent pages.

PROGRAM NO. N5K A

1–10	11–20	21–30	31–40	41–50	51–60	61–70	SEQ. NO. (77–80)
		HEADER				IBFS	
N5KA EXAMPLE CASE						0	01
M	A	S	$\bar{\varepsilon}$	XM	δ ε NP NB NK	MK1 MK2 N1 N2 N3 N4	
0.85	6.4	3.0	1.5	2.0	1 0 3 2 1	0 0 1 1 0 0	02
		k_r					
0.5							03
X_1	X_2	X_3	X_4	Y_1	Y_2		
2.0	4.0	2.5	4.0	0.86603	2.0		04
Z_1	Z_2	NC	NS	ASSØCIATED BØDIES			
0.5	0.5	2	2	1			05
		θ					
0.0	0.5	1.0					06
		τ					
0.0	0.5	1.0					07
X_1	X_2	X_3	X_4	Y_1	Y_2		
2.5	4.0	3.0	4.0	2.0	3.0		08
Z_1	Z_2	NC	NS	ASSØCIATED BØDIES			
0.5	1.0	2	2	0			09
		θ					
0.0	0.5	1.0					10
		τ					
0.0	0.5	1.0					11

PROGRAM NO. N5K A (columns 73 74 75 76)

Cols 1–10	Cols 11–20	Cols 21–30	Cols 31–40	Cols 41–50	Cols 51–60	Cols 61–70	SEQ. NO. (77–80)
X_1	X_2	X_3	X_4	Y_1	Y_2		
2.5	4.0	2.5	4.0	2.0	2.0		12
Z_1	Z_2	NC	NS	ASSØCIATED BØDIES			
0.5	0.0	2	1	2			13
		θ					
0.0	0.5	1.0					14
		τ					
0.0	1.0						15
Z_c	Y_c	a	AR	NBE NSBE NZY NRI	NRS NSH NT1 NT2		
0.0	0.0	1.0	1.0	3 5 1 0	1 0 6 4		16
		ξI					
0.0	2.0	4.0	6.0				17
		ξS					
0.0	1.0	2.0	4.0	5.0	6.0		18
		RS					
0.0	0.5	1.0	1.0	0.5	0.0		19
		$\theta 1$					
0.0	60.0	120.0	180.0	240.0	300.0		20
		$\theta 2$					
0.0	90.0	180.0	270.0				21
FIRST	LAST	. . .	ELEMENTS	FØR $\theta 1$			
2	2						22

PROGRAM NO. N5K A (columns 73–76)

1–10	11–20	21–30	31–40	41–50	51–60	61–70	SEQ. NO. (77–80)
Z_c	Y_c	a	AR	NBE NSBE NZY NRI	NRS NSH NTI NTZ		
-0.5	2.0	0.5	1.0	4 5 2 0	1 0 4 0		23
		ξI					
2.0	2.5	3.25	4.0	4.5			24
		ξS					
2.0	2.25	2.5	3.25	4.5			25
		RS					
0.0	0.25	0.5	0.5	0.0			26
		θI					
45.0	135.0	225.0	315.0				27
FIRST	LAST	...	ELEMENTS	FOR θI			
1	4						28
NQ NRZ N	$az_q^{(b)}$	NQ NRZ N	$az_q^{(b)}$	... modal data for z-bodies			
AZ 1 1 0	1.0	1 2 0	1.0				29
AZ 2 1 1	1.0	2 2 1	1.0				30
AZ 3 2 0	1.666666						31
NQ NRY N	$ay_q^{(b)}$	NQ NRY N		... modal data for y-bodies			
AY 3 2 0	0.333333						32
NQ NRP M N NB	$a_q^{(p)}$	NQ NRP M N NB	$a_q^{(p)}$	... modal data for panels			
A 1 1 0 0 0	1.0	1 2 0 0 0	0.86603				33

PROGRAM NO. N 5 K A

Col 1–2	NQ	NRP	M	N	NB	$a_q^{(p)}$	NQ	NRP	M	N	NB	$a_q^{(p)}$	...modal data for panels	SEQ. NO.
A	2	1	0	1	0	1.0	2	2	0	1	0	0.86603		34
A	3	1	0	0	1	0.577353	3	1	1	0	1	1.0		35
A	3	2	0	0	1	1.321373	3	2	1	0	1	1.0		36
A	3	3	0	0	1	-0.333333	3	3	1	0	1	1.0		37
-1														38

2.4 Example Case Output

```
                              NSKA EXAMPLE CASE

                    **  ARRAY OF REDUCED FREQUENCIES  **

                   0.500000

REFERENCE CHORD =            1.50000
REFERENCE SEMI-SPAN =        3.00000
REFERENCE AREA =             6.00000
MACH NUMBER =                0.35000
MOMENT AXIS =                2.00000
SYMMETRY FLAGS  **  DELTA =  1  EPSILON =  0
NUMBER OF PANELS =   3
NUMBER OF BODIES =   2


                    **  PANEL NO.    1 INPUT VALUES  **


X1 =    2.000000  X2 =    4.000000  Y1 =    0.866030  Z1 =    0.500000

X3 =    2.500000  X4 =    4.000000  Y2 =    2.000000  Z2 =    0.500000

NC =  2  NS =  2

ASSOCIATED BODIES =    1    0    0    0    0    0
                         3  CHORDWISE DIVISIONS FOR PANEL  1


      0.0              0.50000000E 00     0.10000000E 01
                         3  SPANWISE  DIVISIONS FOR PANEL  1


      0.0              0.50000000E 00     0.10000000E 01
                    **  PANEL NO.    2 INPUT VALUES  **


X1 =    2.500000  X2 =    4.000000  Y1 =    2.000000  Z1 =    0.500000

X3 =    3.000000  X4 =    4.000000  Y2 =    3.000000  Z2 =    1.000000

NC =  2  NS =  2

ASSOCIATED BODIES =    0    0    0    0    0    0
                         3  CHORDWISE DIVISIONS FOR PANEL  2


      0.0              0.50000000E 00     0.10000000E 01
                         3  SPANWISE  DIVISIONS FOR PANEL  2


      0.0              0.50000000E 00     0.10000000E 01
                    **  PANEL NO.    3 INPUT VALUES  **


X1 =    2.500000  X2 =    4.000000  Y1 =    2.000000  Z1 =    0.500000

X3 =    2.500000  X4 =    4.000000  Y2 =    2.000000  Z2 =    0.0

NC =  2  NS =  1

ASSOCIATED BODIES =    2    0    0    0    0    0
                         3  CHORDWISE DIVISIONS FOR PANEL  3


      0.0              0.50000000E 00     0.10000000E 01
                         2  SPANWISE  DIVISIONS FOR PANEL  3


      0.0              0.10000000E 01
```

** SUMMARY OF PANEL DATA **

PANEL	NC	NS	NB-ARRAY	DIHEDRAL ANGLE	NO. OF ASSOC. BODIES	LIST OF ASSOCIATED BODIES
1	2	2	4	0.0	1	1
2	2	2	8	26.56505	0	2
3	2	1	10	270.00000	1	2

** GEOMETRY ARRAYS FOR ALL PANELS **

PANEL NO.	STRIP NO.	BOX NO.	3/4 CHORD X	1/4 CHORD X-COORDINATES INBOARD	CENTER	OUTBOARD	BOX CHORD DELTA-X	1/4 CHORD SWEEP ANGLE
1	1	1	2.82813	2.25000	2.35938	2.46875	0.93750	0.36822
1	1	2	3.76563	3.25000	3.29688	3.34375	0.93750	0.16387
1	2	3	2.98438	2.46875	2.57813	2.68750	0.81250	0.36822
1	2	4	3.79688	3.34375	3.39063	3.43750	0.81250	0.16387
2	3	5	3.14063	2.68750	2.79688	2.90625	0.68750	0.37299
2	3	6	3.82813	3.43750	3.48438	3.53125	0.68750	0.16616
2	4	7	3.29688	2.90625	3.01563	3.12500	0.56250	0.37299
2	4	8	3.85938	3.53125	3.57813	3.62500	0.56250	0.16616
3	5	9	3.06250	2.68750	2.68750	2.68750	0.75000	0.0
3	5	10	3.81250	3.43750	3.43750	3.43750	0.75000	0.0

STRIP NO.	Y	Z	DELTA-Y	DELTA-Z	F	CHORD	X-L.E.
1	1.14952	0.50000	0.56698	0.0	0.28349	1.87500	2.12500
2	1.71651	0.50000	0.56698	0.0	0.28349	1.62500	2.37500
3	2.25000	0.62500	0.50000	0.25000	0.27951	1.37500	2.62500
4	2.75000	0.87500	0.50000	0.25000	0.27951	1.12500	2.87500
5	2.00000	0.25000	0.0	-0.50000	0.25000	1.50000	2.50000

```
        **  BODY NO.   1  INPUT VALUES  **

CENTER OF BODY COORDINATES   Y =        0.0          Z =        0.0
AVERAGE HALF-WIDTH OF BODY  =        1.000000
CROSS-SECTIONAL ASPECT RATIO =       1.000000
NUMBER OF INTERFERENCE ELEMENTS ON BODY =   3
NUMBER OF SLENDER BODY ELEMENTS =   5
Z-Y ORIENTATION FLAG =   1
RI-FLAG =  0  R-S FLAG =  1
NUMBER OF DELTA-ETA DELTA-ZETA PAIRS =   0


                     4  XI-I ELEMENTS FOR BODY  1


0.0               0.20000000E 01   0.40000000E 01   0.60000000E 01
                     4  R-I  ELEMENTS FOR BODY  1


0.10000000E 01   0.10000000E 01   0.10000000E 01   0.10000000E 01
                     6  XI-S ELEMENTS FOR BODY  1


0.0               0.10000000E 01   0.20000000E 01   0.40000000E 01   0.50000000E 01   0.60000000E 01
                     6  R-S  ELEMENTS FOR BODY  1


0.0               0.50000000E 00   0.10000000E 01   0.10000000E 01   0.50000000E 00   0.0
                     6  THETA-1 ELEMENTS FOR BODY  1


0.0               0.60000000E 02   0.12000000E 03   0.18000000E 03   0.24000000E 03   0.30000000E 03
                     4  THETA-2 ELEMENTS FOR BODY  1


0.0               0.90000000E 02   0.18000000E 03   0.27000000E 03
                  THE FIRST AND LAST BODY ELEMENTS FOR THETA-1  ON BODY  1


                     2   2   0   0   0   0
```

** BODY NO. 2 INPUT VALUES **

```
CENTER OF BODY COORDINATES   Y =       2.000000      Z =      -0.500000
AVERAGE HALF-WIDTH OF BODY  =        0.500000
CROSS-SECTIONAL ASPECT RATIO =       1.000000
NUMBER OF INTERFERENCE ELEMENTS ON BODY =  4
NUMBER OF SLENDER BODY ELEMENTS =  4
Z-Y ORIENTATION FLAG =  2
PI-FLAG = 0  R-S FLAG = 1
NUMBER OF DELTA-ETA DELTA-ZETA PAIRS =  0
```

5 XI-I ELEMENTS FOR BODY 2

0.20000000E 01	0.25000000E 01	0.32500000E 01	0.40000000E 01	0.45000000E 01

5 R-I ELEMENTS FOR BODY 2

0.50000000E 00	0.50000000E 00	0.50000000E 00	0.50000000E 00	0.50000000E 00

5 XI-S ELEMENTS FOR BODY 2

0.20000000E 01	0.22500000E 01	0.25000000E 01	0.32500000E 01	0.45000000E 01

5 R-S ELEMENTS FOR BODY 2

0.0	0.25000000E 00	0.50000000E 00	0.50000000E 00	0.0

4 THETA-I ELEMENTS FOR BODY 2

0.45000000E 02	0.13500000E 03	0.22500000E 03	0.31500000E 03

THE FIRST AND LAST BODY ELEMENTS FOR THETA-I ON BODY 2

1 4 0 0 0 0

```
** RECEIVING POINT ARRAYS  X, Y, Z, GAMMA **

 0.28281250E 01   0.37656250E 01   0.29843750E 01   0.37968750E 01   0.31406250E 01   0.38281250E 01
 0.32968750E 01   0.38593750E 01   0.30625000E 01   0.38125000E 01   0.10000000E 01   0.30000000E 01
 0.50000000E 01   0.22500000E 01   0.28750000E 01   0.36250000E 01   0.42500000E 01
 0.11495218E 01   0.17165070E 01   0.22500000E 01   0.27500000E 01   0.20000000E 01   0.0
 0.20000000E 01
 0.50000000E 00   0.50000000E 00   0.62500000E 00   0.87500000E 00   0.25000000E 00   0.0
-0.50000000E 00
 0.0              0.0              0.46364743E 00   0.46364743E 00   0.47123871E 01   0.0
 0.0

** SENDING POINT ARRAYS X11, X12, A0, A0-PRIME **

 0.0              0.10000000E 01   0.20000000E 01   0.40000000E 01   0.50000000E 01   0.20000000E 01
 0.22500000E 01   0.25000000E 01   0.32500000E 01
 0.10000000E 01   0.20000000E 01   0.40000000E 01   0.50000000E 01   0.60000000E 01   0.22500000E 01
 0.25000000E 01   0.32500000E 01   0.45000000E 01
 0.25000000E 00   0.75000000E 00   0.10000000E 01   0.75000000E 00   0.25000000E 00   0.12500000E 00
 0.37500000E 00   0.50000000E 00   0.75000000E 00
 0.50000000E 00   0.50000000E 00   0.0             -0.50000000E 00  -0.50000000E 00   0.10000000E 01
 0.10000000E 01   0.0             -0.39999998E 00
```

```
*****

EXECUTION TIME(MINUTES)
CPU             0.0067
I/O             0.0186
TOTAL           0.0253

*****

EXECUTION TIME(MINUTES)
CPU             0.0
I/O             0.0
TOTAL           0.0

*****

EXECUTION TIME(MINUTES)
CPU             0.3162
I/O             0.0351
TOTAL           0.3512
```

```
 INPUT MODAL DATA FOR -- A  --
  1001000      1.0000   1002000        0.8660   2001010       1.0000   2002010       0.8660
  3001001      0.5774   3001101        1.0000   3002001       1.3214   3002101       1.0000
  3003001     -0.3333   3003101        1.0000



 INPUT MODAL DATA FOR -- AZ --
  1001000      1.0000   1002000        1.0000   2001100       1.0000   2002100       1.0000
  3002000      1.6667



 INPUT MODAL DATA FOR -- AY --
  3002000      0.3333

*****

EXECUTION TIME(MINUTES)
CPU               0.0000
I/O               0.0062
TOTAL             0.0062

*****

EXECUTION TIME(MINUTES)
CPU               0.0
I/O               0.0
TOTAL             0.0

CALCULATION OF DZ AND DY.  N =    21 NM =      3 NTZS =      9 NTYS =      4
                                MAX. CORE NEEDED =    320 CORE AVAIL. = 10000

 ALL CALCULATIONS DONE IN CORE.

  MODE -          1
  NTZS -          9
  NTYS -          4
  U-Z, U-Y, CP-Z*DELTA-A, CP-Y*DELTA-A

 0.0         0.7854E 00  0.0         0.7069E 01  0.0         0.1257E 02
 0.0         0.7069E 01  0.0         0.7854E 00  0.0         0.1963E 00
 0.0         0.1767E 01  0.0         0.3142E 01  0.0         0.7854E 00
 0.0         0.0         0.0         0.0         0.0         0.0
 0.0         0.0        -0.5236E 00  0.3142E 01 -0.4712E 01  0.9425E 01
-0.1676E 02  0.0        -0.4712E 01 -0.9425E 01 -0.5236E 00 -0.3142E 01
-0.3272E-01  0.7854E 00 -0.2945E 00  0.2356E 01 -0.1571E 01  0.0
-0.6545E 00 -0.3142E 01  0.0         0.0         0.0         0.0
 0.0         0.0         0.0         0.0

  MODE -          2
  NTZS -          9
  NTYS -          4
  U-Z, U-Y, CP-Z*DELTA-A, CP-Y*DELTA-A

 0.7854E 00  0.2618E 00  0.7069E 01  0.7069E 01  0.1257E 02  0.2513E 02
 0.7069E 01  0.2121E 02  0.7854E 00  0.2880E 01  0.1963E 00  0.2782E 00
 0.1767E 01  0.2798E 01  0.3142E 01  0.6021E 01  0.7854E 00  0.2029E 01
 0.0         0.0         0.0         0.0         0.0         0.0
 0.0         0.0         0.2967E 01  0.2094E 01  0.4712E 01  0.1885E 02
-0.3351E 02  0.3351E 02 -0.2356E 02 -0.1885E 02 -0.5061E 01 -0.1047E 02
 0.7399E 00  0.1178E 01  0.1890E 01  0.4320E 01 -0.3011E 01  0.3142E 01
-0.4832E 01 -0.6807E 01  0.0         0.0         0.0         0.0
 0.0         0.0         0.0         0.0

  MODE -          3
  NTZS -          9
  NTYS -          4
  U-Z, U-Y, CP-Z*DELTA-A, CP-Y*DELTA-A

 0.0         0.0         0.0         0.0         0.0         0.0
 0.0         0.0         0.0         0.0         0.0         0.3272E 00
 0.0         0.2945E 01  0.0         0.5236E 01  0.0         0.1309E 01
 0.0         0.6545E-01  0.0         0.5890E 00  0.0         0.1047E 01
 0.0         0.2618E 00  0.0         0.0         0.0         0.0
 0.0         0.0         0.0         0.0         0.0         0.0
-0.5454E-01  0.1309E 01 -0.4909E 00  0.3927E 01 -0.2618E 01  0.0
-0.1091E 01 -0.5236E 01 -0.1091E-01  0.2618E 00 -0.9817E-01  0.7854E 00
-0.5236E 00  0.0        -0.2182E 00 -0.1047E 01
```

```
COLUMN      1 OF DELTA- W-
   1 -0.16765E 00 -0.26725E 00      2 -0.27663E 00 -0.40265E 00
   3 -0.17858E 00  0.47032E-01      4 -0.22729E 00 -0.17359E 00
   5 -0.16776E 00  0.19570E-01      6 -0.13306E 00 -0.10716E 00
   7 -0.12543E 00 -0.63007E-02      8 -0.14421E 00 -0.46031E-01
   9  0.21474E 01  0.56584E-01     10  0.27783E-01  0.76223E-01
  11  0.75136E-02 -0.13743E-01     12 -0.72854E-01 -0.40619E-01
  13 -0.19596E-02 -0.38105E-01     14 -0.12996E 00 -0.10007E 00
  15 -0.10968E 00 -0.10643E 00     16 -0.17229E 00 -0.11741E 00
  17 -0.17035E 00 -0.18190E 00     18 -0.20427E-01 -0.12301E 00
  19 -0.13418E-01 -0.10852E 00     20 -0.44763E-01 -0.12254E 00
  21 -0.91678E-02 -0.13222E 00

COLUMN      2 OF DELTA- W-
   1 -0.59579E 00 -0.29402E 00      2 -0.10651E 01 -0.45037E 00
   3 -0.28245E 00  0.32520E 00      4 -0.69030E 00 -0.86980E-02
   5 -0.27637E 00  0.25650E 00      6 -0.50948E 00  0.58693E-01
   7 -0.22315E 00  0.15608E 00      8 -0.33829E 00  0.11830E 00
   9  0.10460E 00  0.80871E-01     10  0.15617E 00  0.12543E 00
  11 -0.19271E-02 -0.33447E-01     12 -0.18143E 00  0.56348E-02
  13 -0.44642E-01 -0.81319E-01     14 -0.33736E 00  0.27377E-02
  15 -0.30299E 00 -0.57571E-01     16 -0.47061E 00  0.14893E-01
  17 -0.60529E 00 -0.16253E 00     18 -0.18410E 00 -0.18497E 00
  19 -0.14730E 00 -0.18584E 00     20 -0.24727E 00 -0.19016E 00
  21 -0.19684E 00 -0.29960E 00

COLUMN      3 OF DELTA- W-
   1 -0.77452E-01  0.78302E-01      2 -0.67830E-01 -0.57944E-01
   3 -0.92054E-0.  0.32170E 00      4 -0.64814E-01  0.21993E-01
   5 -0.84912E-01  0.23316E 00      6 -0.42804E-01  0.69834E-01
   7 -0.52960E-01  0.70497E-01      8 -0.37714E-01  0.29039E-01
   9  0.81054E-02 -0.13012E 00     10 -0.15097E-01 -0.11022E 00
  11  0.17523E-01 -0.22905E-01     12 -0.12142E 00 -0.67699E-01
  13 -0.32660E-02 -0.63509E-01     14  0.12716E-01  0.96548E-02
  15 -0.29452E-02  0.18996E-01     16 -0.17280E-01  0.10752E-01
  17 -0.19371E-01  0.14849E-02     18  0.13209E-01  0.39783E-02
  19  0.11211E-01  0.64860E-02     20  0.13454E-01  0.10831E-01
  21  0.97084E-02  0.17423E-01

*****

EXECUTION TIME(MINUTES)
CPU                 0.0754
I/O                 0.0404
TOTAL               0.1158
ABSOLUTE COORDINATES USED FOR PANEL     1.
ABSOLUTE COORDINATES USED FOR PANEL     2.

W FOR MODE      1

 0.0          0.1000E 01  0.0          0.1000E 01  0.0          0.1000E 01
 0.0          0.1000E 01  0.0          0.8660E 00  0.0          0.8660E 00
 0.0          0.8660E 00  0.0          0.8660E 00  0.0          0.0
 0.0          0.0

--WI-- FOR MODE     1

 0.1676E 00   0.1267E 01   0.2766E 00   0.1403E 01   0.1786E 00   0.9530E 00
 0.2273E 00   0.1174E 01   0.1628E 00   0.8465E 00   0.1831E 00   0.9732E 00
 0.1252E 00   0.8723E 00   0.1542E 00   0.9121E 00  -0.2147E-01  -0.5658E-01
-0.2776E-01  -0.7622E-01  -0.7514E-02   0.1374E-01   0.7285E-01   0.4062E-01
 0.1960E-02   0.3811E-01   0.1300E 00   0.1010E 00   0.1097E 00   0.1064E 00
 0.1723E 00   0.1174E 00   0.1703E 00   0.1819E 00   0.2043E-01   0.1230E 00
 0.1347E-01   0.1025E 00   0.4476E-01   0.1225E 00   0.9168E-02   0.1322E 00
ABSOLUTE COORDINATES USED FOR PANEL     1.
ABSOLUTE COORDINATES USED FOR PANEL     2.
```

```
W FOR MODE       2

 0.1000E 01  0.1885E 01  0.1000E 01  0.2510E 01  0.1000E 01  0.1990E 01
 0.1000E 01  0.2531E 01  0.8660E 00  0.1813E 01  0.8660E 00  0.2210E 01
 0.8660E 00  0.1903E 01  0.8660E 00  0.2228E 01  0.0         0.0
 0.0         0.0

--WT-- FOR MODE     2

 0.1596E 01  0.2179E 01  0.2065E 01  0.2961E 01  0.1282E 01  0.1664E 01
 0.1690E 01  0.2540E 01  0.1142E 01  0.1557E 01  0.1376E 01  0.2151E 01
 0.1089E 01  0.1747E 01  0.1204E 01  0.2110E 01 -0.1046E 00 -0.8087E-01
-0.1561E 00 -0.1254E 00  0.1927E-02  0.3345E-01  0.1814E 00 -0.5635E-02
 0.4464E-01  0.8132E-01  0.3374E 00 -0.2738E-02  0.3030E 00  0.5757E-01
 0.4706E 00 -0.1489E-01  0.6053E 00  0.1625E 00  0.1841E 00  0.1850E 00
 0.1473E 00  0.1859E 00  0.2473E 00  0.1902E 00  0.1968E 00  0.2996E 00
RELATIVE COORDINATES USED FOR PANEL      1.
RELATIVE COORDINATES USED FOR PANEL      2.
RELATIVE COORDINATES USED FOR PANEL      3.
```

```
W FOR MODE       3

 0.0         0.7663E 00  0.0         0.7663E 00  0.0         0.1144E 01
 0.0         0.1144E 01  0.0         0.1508E 01  0.0         0.1508E 01
 0.0         0.1880E 01  0.0         0.1880E 01  0.0        -0.1667E 00
 0.0        -0.1667E 00

--WT-- FOR MODE     3

 0.7745E-01  0.6880E 00  0.6783E-01  0.8243E 00  0.9205E-01  0.8226E 00
 0.6481E-01  0.1122E 01  0.8491E-01  0.1275E 01  0.4280E-01  0.1438E 01
 0.5296E-01  0.1810E 01  0.3771E-01  0.1851E 01 -0.8105E-02 -0.3655E-01
 0.1510E-01 -0.5644E-01 -0.1252E-01  0.2291E-01  0.1214E 00  0.6770E-01
 0.3266E-02  0.6351E-01 -0.1272E-01 -0.9655E-02  0.2945E-02 -0.1900E-01
 0.1728E-01 -0.1075E-01  0.1937E-01 -0.1485E-02 -0.1321E-01 -0.3978E-02
-0.1121E-01 -0.6486E-02 -0.1345E-01 -0.1083E-01 -0.9708E-02 -0.1742E-01

*****

EXECUTION TIME(MINUTES)
CPU                  0.0133
I/O                  0.0106
TOTAL                0.0240

*****

EXECUTION TIME(MINUTES)
CPU                  0.0111
I/O                  0.0133
TOTAL                0.0244
```

COLUMN NO. 1 OF GAMMAS FOLLOWS

0.19665623E 01	0.42423382E 01	-0.11037083E 01	0.46231680E 01	0.26930857E 01	0.48133154E 01
-0.17455091E 01	0.49378624E 01	0.18580713E 01	0.43687849E 01	-0.11078661E 01	0.30424414E 01
0.16848707E 01	0.43148623E 01	-0.14925518E 01	0.19979200E 01	0.13595495E 01	0.10761557E 01
-0.76321185E-01	0.16339111E 01	0.57387359E-01	0.15478647E 00	0.64842226E 00	-0.33000479E 01
-0.54306471E 00	-0.96502858E 00	0.30278474E 00	0.16926819E 00	-0.449[illegible]847E-0[illegible]	-0.56636047E 00
0.41055709E 00	-0.86979937E 00	-0.12379962E 00	-0.16611433E [illegible]	0.2414[illegible]331E 00	0.40218151E 00
0.10244246E 01	0.94769959E 00	0.13733473E 01	0.15477180E 01	0.14319327E 01	0.99667245E 00

COLUMN NO. 2 OF GAMMAS FOLLOWS

0.83234043E 01	0.55460453E 01	0.36448412E 01	0.12989089E 02	0.10454334E 02	0.62213135E 01
0.24164743E 01	0.14484083E 02	0.84354038E 01	0.69312830E 01	-0.34786558E 00	0.10142653E 02
0.79901028E 01	0.73800545E 01	-0.99627325E 00	0.72473288E 01	0.37644078E 01	0.34082592E 00
0.20209637E 01	0.39445601E 01	0.29087239E 00	0.11433595E 00	-0.24475050E 01	-0.75162716E 01
-0.24893293E 01	-0.19528704E 01	0.65337712E 00	-0.14427942E 00	-0.72343698E 00	-0.12275944E 01
-0.10911363E 00	-0.28704632E 01	-0.31925070E 00	-0.55605429E 00	0.10812321E 01	0.29080349E 00
0.29690113E 01	0.55575371E 00	0.47090960E 01	0.19422626E 01	0.47681456E 01	0.82420814E 00

COLUMN NO. 3 OF GAMMAS FOLLOWS

0.17746496E 01	0.27028809E 01	-0.95816278E 00	0.40935907E 01	0.21796160E 01	0.41750998E 01
-0.21127090E 01	0.47413349E 01	0.16989136E 01	0.62863846E 01	-0.30868311E 01	0.36716051E 01
0.13319178E 01	0.77920647E 01	-0.30928459E 01	0.24183979E 01	0.71831417E 00	-0.43016274E-01
0.19243479E 00	0.88363409E 00	0.85615396E-01	0.19928098E 00	0.15352192E 01	-0.21076336E 01
-0.38684684E 00	-0.39633286E 00	-0.15489656E 00	-0.13395346E 00	-0.25440836E 00	-0.97130507E 00
0.69175780E-01	-0.13086824E 01	-0.44596702E 00	-0.89562064E 00	0.20327300E 00	-0.15731261E-02
0.51055562E 00	0.27052313E 00	0.47117102E 00	0.48530942E 00	0.61559707E 00	0.19583944E-01

THE 21 X 21 MATRIX WITH 3 RIGHT SIDES WAS SOLVED DIRECTLY IN 0.005 MINUTES.

*****

EXECUTION TIME(MINUTES)
CPU 0.0211
I/O 0.0146
TOTAL 0.0357

LOADS ON SLENDER BODY ELEMENTS DUE TO LIFTING SURFACE BOXES.

	FZ			FY	MZ			MY
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	0.5731E 00	0.7088E 01	0.0	0.0	0.1142E 01	-0.9615E 00	0.0	0.0
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

LOADS ON SLENDER BODY ELEMENTS DUE TO LIFTING SURFACE BOXES AND INTERFERENCE BODY ELEMENTS.

	FZ			FY	MZ			MY
1	0.2869E-01	0.7739E-01	0.0	0.0	0.0	0.0	0.0	0.0
2	0.2869E-01	0.7739E-01	0.0	0.0	0.0	0.0	0.0	0.0
3	0.5476E 00	0.6812E 01	0.0	0.0	0.1246E 01	-0.9483E 00	0.0	0.0
4	-0.4099E 00	0.7298E 00	0.0	0.0	-0.2773E-01	0.3652E-01	0.0	0.0
5	-0.2980E 00	0.5838E 00	0.0	0.0	0.0	0.0	0.0	0.0

ELEMENT LOADS FOR SLENDER BODY NUMBER 1. MODE NO. 1

	FZ			FY	MZ			MY
1	-0.2331E 00	0.1648E 01	0.0	0.0	0.0	0.0	0.0	0.0
2	-0.2327E 01	0.4790E 01	0.0	0.0	0.0	0.0	0.0	0.0
3	-0.8306E 01	0.7073E 01	0.0	0.0	0.1212E 01	-0.2553E 00	0.0	0.0
4	-0.2819E 01	-0.4243E 01	0.0	0.0	-0.4132E-01	-0.2870E-01	0.0	0.0
5	-0.5598E 00	-0.9870E 00	0.0	0.0	0.0	0.0	0.0	0.0

LOADS ON SLENDER BODY ELEMENTS DUE TO LIFTING SURFACE BOXES.

	FZ			FY	MZ			MY
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	0.9193E 01	0.1513E 02	0.0	0.0	0.9295E 00	-0.4479E 01	0.0	0.0
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

LOADS ON SLENDER BODY ELEMENTS DUE TO LIFTING SURFACE BOXES AND INTERFERENCE BODY ELEMENTS.

	FZ			FY	MZ			MY
1	0.1454E 00	0.5717E-01	0.0	0.0	0.0	0.0	0.0	0.0
2	0.1454E 00	0.5717E-01	0.0	0.0	0.0	0.0	0.0	0.0
3	0.8732E 01	0.1435E 02	0.0	0.0	0.1097E 01	-0.4602E 01	0.0	0.0
4	-0.5408E-01	0.1861E 01	0.0	0.0	-0.1091E-01	0.1175E 00	0.0	0.0
5	-0.1046E-01	0.1391E 01	0.0	0.0	0.0	0.0	0.0	0.0

ELEMENT LOADS FOR SLENDER BODY NUMBER 1. MODE NO. 2

	FZ			FY	MZ			MY
1	0.1629E 01	0.1104E 01	0.0	0.0	0.0	0.0	0.0	0.0
2	0.2502E 01	0.9482E 01	0.0	0.0	0.0	0.0	0.0	0.0
3	-0.8794E 01	0.3205E 02	0.0	0.0	0.1774E 01	-0.3216E 01	0.0	0.0
4	-0.1224E 02	-0.8129E 01	0.0	0.0	-0.1112E 00	-0.2379E-01	0.0	0.0
5	-0.2541E 01	-0.3845E 01	0.0	0.0	0.0	0.0	0.0	0.0

LOADS ON SLENDER BODY ELEMENTS DUE TO LIFTING SURFACE BOXES.

	FZ		FY		MZ		MY	
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	0.1804E 00	0.6356E 01	0.0	0.0	0.1214E 01	-0.1200E 01	0.0	0.0
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

LOADS ON SLENDER BODY ELEMENTS DUE TO LIFTING SURFACE BOXES AND INTERFERENCE BODY ELEMENTS.

	FZ		FY		MZ		MY	
1	0.4281E-01	0.9964E-01	0.0	0.0	0.0	0.0	0.0	0.0
2	0.4281E-01	0.9964E-01	0.0	0.0	0.0	0.0	0.0	0.0
3	0.1335E-02	0.6022E 01	0.0	0.0	0.1181E 01	-0.1277E 01	0.0	0.0
4	-0.5875E 00	0.5136E 00	0.0	0.0	-0.2674E-01	0.2144E-01	0.0	0.0
5	-0.4805E 00	0.4278E 00	0.0	0.0	0.0	0.0	0.0	0.0

ELEMENT LOADS FOR SLENDER BODY NUMBER 1. MODE NO. 3

	FZ		FY		MZ		MY	
1	0.4281E-01	0.9964E-01	0.0	0.0	0.0	0.0	0.0	0.0
2	0.4281E-01	0.9964E-01	0.0	0.0	0.0	0.0	0.0	0.0
3	-0.7913E 00	0.6456E 01	0.0	0.0	0.1125E 01	-0.1217E 00	0.0	0.0
4	-0.6781E 00	0.7876E-01	0.0	0.0	-0.4938E-01	-0.8727E-01	0.0	0.0
5	-0.4805E 00	0.4278E 00	0.0	0.0	0.0	0.0	0.0	0.0

LOADS ON SLENDER BODY ELEMENTS DUE TO LIFTING SURFACE BOXES.

	FZ		FY		MZ		MY	
1	-0.8696E-03	-0.1876E-02	0.1507E-01	0.3250E-01	0.5435E-04	0.1172E-03	-0.9417E-03	-0.2031E-02
2	-0.1760E-01	-0.3209E-01	0.4533E-01	0.9217E-01	0.9910E-03	0.1771E-02	-0.9499E-03	-0.1698E-02
3	-0.1263E 00	-0.2681E 00	0.2013E 00	0.3089E 00	-0.2897E-02	0.5300E-02	0.2062E-01	0.1438E-01
4	0.9830E-01	-0.2649E 00	-0.5555E-01	0.3871E 00	0.2221E-01	-0.6709E-01	-0.2193E-01	0.9212E-01

LOADS ON SLENDER BODY ELEMENTS DUE TO LIFTING SURFACE BOXES AND INTERFERENCE BODY ELEMENTS.

	FZ		FY		MZ		MY	
1	0.1519E 00	0.8226E-01	0.1879E 00	0.2358E 00	0.5435E-04	0.1172E-03	-0.9417E-03	-0.2031E-02
2	0.1352E 00	0.5205E-01	0.2182E 00	0.2954E 00	0.9910E-03	0.1771E-02	-0.9499E-03	-0.1698E-02
3	-0.1019E 00	-0.5373E 00	0.4617E 00	0.6071E 00	-0.2897E-02	0.5300E-02	0.2062E-01	0.1438E-01
4	-0.2140E 00	0.2037E 00	0.7197E-01	0.1046E 00	0.1186E 00	-0.1827E 00	-0.6059E-02	0.2040E 00

ELEMENT LOADS FOR SLENDER BODY NUMBER 2. MODE NO. 1

	FZ		FY		MZ		MY	
1	0.8094E-01	0.4811E 00	0.1580E 00	0.2358E 00	0.5435E-04	0.1172E-03	-0.9417E-03	-0.2031E-02
2	-0.6873E-01	0.1249E 01	0.1892E 00	0.2954E 00	0.9910E-03	0.1771E-02	-0.9499E-03	-0.1698E-02
3	-0.1063E 01	-0.5373E 00	0.3748E 00	0.6071E 00	-0.2897E-02	0.5300E-02	0.2062E-01	0.1438E-01
4	-0.7706E 00	-0.1514E 01	-0.4761E-01	0.3935E-01	0.1007E 00	-0.1369E 00	-0.1557E-01	0.2285E 00

LOADS ON SLENDER BODY ELEMENTS DUE TO LIFTING SURFACE BOXES.

	FZ		FY		MZ		MY	
1	-0.3680E-02	-0.2452E-02	0.6377E-01	0.4249E-01	0.2300E-03	0.1533E-03	-0.3986E-02	-0.2656E-02
2	-0.6892E-01	-0.4153E-01	0.1865E 00	0.1201E 00	0.3847E-02	0.2289E-02	-0.3688E-02	-0.2194E-02
3	-0.5393E 00	-0.3938E 00	0.7260E 00	0.3489E 00	0.8394E-03	0.1785E-01	0.5261E-01	-0.6045E-02
4	-0.1137E 00	-0.7851E 00	0.4096E 00	0.1012E 01	-0.3410E-01	-0.1942E 00	0.8107E-01	0.2557E 00

LOADS ON SLENDER BODY ELEMENTS DUE TO LIFTING SURFACE BOXES AND INTERFERENCE BODY ELEMENTS.

	FZ		FY		MZ		MY	
1	0.3236E 00	-0.7907E-01	0.6104E 00	0.1844E 00	0.2300E-03	0.1533E-03	-0.3986E-02	-0.2656E-02
2	0.2584E 00	-0.1182E 00	0.7332E 00	0.2660E 00	0.3847E-02	0.2289E-02	-0.3688E-02	-0.2194E-02
3	-0.7464E 00	-0.1083E 01	0.1640E 01	0.7629E 00	0.8394E-03	0.1785E-01	0.5261E-01	-0.6045E-02
4	-0.1315E 00	0.5613E 00	0.5082E 00	0.5264E 00	0.2832E-01	-0.5989E 00	0.2458E 00	0.4771E 00

ELEMENT LOADS FOR SLENDER BODY NUMBER 2. MODE NO. 2

	FZ		FY		MZ		MY	
1	0.5902E 00	0.6279E 00	0.5525E 00	0.2464E 00	0.2300E-03	0.1533E-03	-0.3986E-02	-0.2656E-02
2	0.1109E 01	0.2184E 01	0.6752E 00	0.3240E 00	0.3847E-02	0.2289E-02	-0.3688E-02	-0.2194E-02
3	-0.2601E 01	0.8381E 00	0.1466E 01	0.9368E 00	0.8394E-03	0.1785E-01	0.5261E-01	-0.6045E-02
4	-0.3217E 01	-0.2814E 01	0.1712E 00	0.5699E 00	0.6144E-01	-0.4256E 00	0.2634E 00	0.5695E 00

LOADS ON SLENDER BODY ELEMENTS DUE TO LIFTING SURFACE BOXES.

	FZ		FY		MZ		MY	
1	-0.7847E-03	-0.1195E-02	0.1360E-01	0.2071E-01	0.4904E-04	0.7470E-04	-0.8498E-03	-0.1294E-02
2	-0.1440E-01	-0.2697E-01	0.3949E-01	0.6498E-01	0.8021E-03	0.1536E-02	-0.7688E-03	-0.1473E-02
3	-0.1022E 00	-0.2735E 00	0.1241E 00	0.3230E-01	-0.2875E-02	0.8202E-02	0.1842E-01	0.2076E-01
4	0.1360E 00	-0.2694E 00	0.2966E-01	0.2340E 00	0.3091E-01	-0.6906E-01	-0.3501E-02	0.4811E-01

LOADS ON SLENDER BODY ELEMENTS DUE TO LIFTING SURFACE BOXES AND INTERFERENCE BODY ELEMENTS.

	FZ		FY		MZ		MY	
1	-0.8021E-01	-0.9558E-01	0.1164E 00	0.1939E-01	0.4904E-04	0.7470E-04	-0.8498E-03	-0.1294E-02
2	-0.9383E-01	-0.1214E 00	0.1423E 00	0.6366E-01	0.8021E-03	0.1536E-02	-0.7688E-03	-0.1473E-02
3	-0.4761E-01	-0.5620E 00	0.1909E 00	0.1544E 00	-0.2875E-02	0.8202E-02	0.1842E-01	0.2076E-01
4	-0.1890E 00	-0.3217E-01	0.1218E 00	-0.6183E-01	0.1210E 00	-0.1478E 00	-0.2215E-01	0.1183E 00

ELEMENT LOADS FOR SLENDER BODY NUMBER 2. MODE NO. 3

	FZ		FY		MZ		MY	
1	-0.1079E 00	0.5691E 00	0.1109E 00	0.1523E 00	0.4904E-04	0.7470E-04	-0.8498E-03	-0.1294E-02
2	-0.3431E 00	0.1873E 01	0.9246E-01	0.4625E 00	0.8021E-03	0.1536E-02	-0.7688E-03	-0.1473E-02
3	-0.1377E 01	-0.5620E 00	-0.7495E-01	0.1544E 00	-0.2875E-02	0.8202E-02	0.1842E-01	0.2076E-01
4	-0.7429E 00	-0.2691E 01	0.1101E-01	-0.5936E 00	0.1210E 00	-0.1478E 00	-0.2215E-01	0.1183E 00

*****

EXECUTION TIME(MINUTES)

CPU	0.0300
I/O	0.0319
TOTAL	0.0619

MODE NO 1

STRIP NO.	Y	Z	YOS	LIFT COEFFICIENT		MOMENT COEFFICIENT		CP-R AND CP-I	
1	1.1495	0.5000	0.3832	0.431427	4.432753	0.329855	-0.601678	-0.51457	0.38574
2	1.7165	0.5000	0.5722	0.473788	4.875587	0.495600	-0.625017	-0.79604	0.37819
3	[illegible].2500	0.6250	0.7500	0.025103	3.705612	0.455104	-0.297408	-17.87962	0.33026
4	2.7500	0.8750	0.9167	0.096159	3.156390	0.385158	-0.104931	-3.75541	0.28324
5	2.0000	0.2500	0.6667	0.641614	1.355033	0.099282	-0.239099	0.09526	0.42645

BODY ELEM.	Y	Z	XOL	LIFT COEFFICIENT		MOMENT COEFFICIENT	
1	0.0	0.0	0.0833	-0.233106	1.648189	0.0	0.0
1	0.0	0.0	0.0833	0.0	0.0	0.0	0.0
2	0.0	0.0	0.2500	-2.327499	4.789781	0.0	0.0
2	0.0	0.0	0.2500	0.0	0.0	0.0	0.0
3	0.0	0.0	0.5000	-4.152803	3.536185	0.605896	-0.127639
3	0.0	0.0	0.5000	0.0	0.0	0.0	0.0
4	0.0	0.0	0.7500	-2.819480	-4.243449	-0.041323	-0.028705
4	0.0	0.0	0.7500	0.0	0.0	0.0	0.0
5	0.0	0.0	0.9167	-0.559796	-0.987041	0.0	0.0
5	0.0	0.0	0.9167	0.0	0.0	0.0	0.0

BODY ELEM.	Y	Z	XOL	MODIFIED LIFT COEFFICIENTS, Z- AND Y-DIRECTIONS			
1	0.0	0.0	0.0833	1.437844	1.588631	0.0	0.0
2	0.0	0.0	0.2500	-2.298344	6.563907	0.0	0.0
3	0.0	0.0	0.5000	-8.822783	12.580199	0.0	0.0
4	0.0	0.0	0.7500	-6.624901	-4.239579	0.0	0.0
5	0.0	0.0	0.9167	-1.623070	-2.175225	0.0	0.0

BODY NO.	Y	Z	X-CENTER	TOTAL LIFT COEFF.		TOTAL MOMENT COEFF.	
1	0.0	0.0	3.0000	-2.225857	1.293789	4.735582	-0.519459
1	0.0	0.0	3.0000	0.0	0.0	0.0	0.0

BODY ELEM.	Y	Z	XOL	LIFT COEFFICIENT		MOMENT COEFFICIENT	
6	2.0000	-0.5000	0.0500	0.323774	1.924397	0.000217	0.000469
6	2.0000	-0.5000	0.0500	0.634755	0.943069	-0.003767	-0.008[illegible]26
7	2.0000	-0.5000	0.1500	-0.[illegible]74939	4.994209	0.003964	0.007085
7	2.0000	-0.5000	0.1500	0.756814	1.191733	-0.003799	-0.006791
8	2.0000	-0.5000	0.3500	-1.4[illegible]6803	-0.716422	-0.003862	0.007066
8	2.0000	-0.5000	0.3500	0.499669	0.809400	0.027492	0.019173
9	2.0000	-0.5000	0.7500	-0.616455	-1.711128	0.080586	-0.109485
9	2.0000	-0.5000	0.7500	-0.035064	0.031481	-0.012456	0.182784

BODY ELEM.	Y	Z	XOL	MODIFIED LIFT COEFFICIENTS, Z- AND Y-DIRECTIONS			
6	2.0000	-0.5000	0.0500	0.740464	2.146836	0.705551	0.560749
7	2.0000	-0.5000	0.1500	-0.314993	3.342005	0.703525	0.833136
8	2.0000	-0.5000	0.3500	-2.580956	-0.578183	0.865402	1.410345
9	2.0000	-0.5000	0.7500	-[illegible].869089	-3.230846	-0.112051	0.158218

BODY NO.	Y	Z	X-CENTER	TOTAL LIFT COEFF.		TOTAL MOMENT COEFF.	
2	2.0000	-0.5000	3.2500	-0.284524	-0.050246	0.259284	0.276117
2	2.0000	-0.5000	3.2500	0.105513	0.184000	-0.033990	-0.052717

CZ =	-2.492867	5.420683	CY =	0.0	0.0
CM =	2.574017	-1.015308	CN =	0.0	0.0
CSL=	0.0	0.0			

MODE NO 2

STRIP NO.	Y	Z	YOS	LIFT COEFFICIENT		MOMENT COEFFICIENT		CP-R AND CP-I	
1	1.1495	0.5000	0.3832	5.984121	9.267561	-0.163195	-2.088826	0.27727	0.47539
2	1.7165	0.5000	0.5722	6.435728	10.352689	0.200348	-2.326933	0.21887	0.47477
3	2.2500	0.6250	0.7500	4.043764	8.536967	0.592437	-1.469541	0.10349	0.42202
4	2.7500	0.8750	0.9167	3.496903	7.313690	0.686186	-0.897620	0.05377	0.37273
5	2.0000	0.2500	0.6667	2.892935	2.142693	-0.143624	-0.718303	0.29965	0.58523

BODY ELEM.	Y	Z	XOL	LIFT COEFFICIENT		MOMENT COEFFICIENT	
1	0.0	0.0	0.0833	1.628965	1.104363	0.0	0.0
1	0.0	0.0	0.0833	.0	0.0	0.0	0.0
2	0.0	0.0	0.2500	2.501630	9.481941	0.0	0.0
2	0.0	0.0	0.2500	0.0	0.0	0.0	0.0
3	0.0	0.0	0.5000	-4.352023	16.024748	0.887208	-1.607799
3	0.0	0.0	0.5000	0.0	0.0	0.0	0.0
4	0.0	0.0	0.7500	-12.226346	-8.129066	-0.111233	-0.023787
4	0.0	0.0	0.7500	0.0	0.0	0.0	0.0
5	0.0	0.0	0.9167	-2.541192	-3.845133	0.0	0.0
5	0.0	0.0	0.9167	0.0	0.0	0.0	0.0

BODY ELEM.	Y	Z	XOL	MODIFIED LIFT COEFFICIENTS, Z- AND Y-DIRECTIONS			
1	0.0	0.0	0.0833	3.875882	-1.427946	0.0	0.0
2	0.0	0.0	0.2500	3.660136	10.856790	0.0	0.0
3	0.0	0.0	0.5000	0.109172	46.309570	0.0	0.0
4	0.0	0.0	0.7500	-22.057449	-2.866598	0.0	0.0
5	0.0	0.0	0.9167	-7.123833	-6.340250	0.0	0.0

BODY NO.	Y	Z	X-CENTER	TOTAL LIFT COEFF.		TOTAL MOMENT COEFF.	
1	0.0	0.0	3.0000	-3.023586	4.790936	9.609210	-5.878701
1	0.0	0.0	3.0000	0.0	0.0	0.0	0.0

BODY ELEM.	Y	Z	XOL	LIFT COEFFICIENT		MOMENT COEFFICIENT	
6	2.0000	-0.5000	0.0500	2.360761	2.511539	0.000920	0.000613
6	2.0000	-0.5000	0.0500	2.209808	0.985413	-0.015943	-0.010623
7	2.0000	-0.5000	0.1500	4.437428	8.736561	0.015389	0.009157
7	2.0000	-0.5000	0.1500	2.700893	1.295815	-0.014750	-0.008777
8	2.0000	0.5000	0.3500	-3.468510	1.117512	0.001119	0.023807
8	2.0000	-0.5000	0.3500	1.954643	1.249096	0.070149	-0.008061
9	2.0000	-0.5000	0.7500	-2.573796	-2.250378	0.049156	-0.340507
9	2.0000	-0.5000	0.7500	0.136960	0.455902	0.210737	0.455594

BODY ELEM.	Y	Z	XOL	MODIFIED LIFT COEFFICIENTS, Z- AND Y-DIRECTIONS			
6	2.0000	-0.5000	0.0500	3.423564	2.662498	1.930327	0.205220
7	2.0000	-0.5000	0.1500	2.955782	6.027204	2.287436	0.718613
8	2.0000	-0.5000	0.3500	-5.456857	3.166936	3.402420	2.092484
9	2.0000	-0.5000	0.7500	-7.399488	-5.741070	0.469216	1.369649

BODY NO.	Y	Z	X-CENTER	TOTAL LIFT COEFF.		TOTAL MOMENT COEFF.	
2	2.0000	-0.5000	3.2500	-0.643606	0.130693	0.821367	0.337447
2	2.0000	-0.5000	3.2500	0.447[illegible]5	0.324532	-0.168504	-0.154367

CZ =	1.013682	14.231640	CY =	0.0	0.0
CM =	4.635420	-5.763388	CN =	0.0	0.0
CSI =	0.0	0.0			

MODE NO 3

STRIP NO.	Y	Z	YOS	LIFT COEFFICIENT		MOMENT COEFFICIENT		CP-R AND CP-I	
1	1.1495	0.5000	0.3812	0.408243	3.398235	0.290571	-0.598618	-0.46176	0.42616
2	1.7165	0.5000	0.5722	0.033454	4.458212	0.532358	-0.628056	-15.66328	0.39088
3	2.2500	0.6250	0.7500	-0.693959	4.978993	0.684963	-0.295527	1.23704	0.30935
4	2.7500	0.3750	0.9167	-0.980463	5.105230	0.663153	0.033554	1.00319	0.24343
5	2.0000	0.2500	0.6667	0.455374	0.420309	0.008813	-0.168370	0.23065	0.65059

BODY ELEM.	Y	Z	XOL	LIFT COEFFICIENT		MOMENT COEFFICIENT	
1	0.0	0.0	0.0833	0.042808	0.099640	0.0	0.0
1	0.0	0.0	0.0833	0.0	0.0	0.0	0.0
2	0.0	0.0	0.2500	0.042808	0.099640	0.0	0.0
2	0.0	0.0	0.2500	0.0	0.0	0.0	0.0
3	0.0	0.0	0.5000	-0.395654	3.228248	0.562377	-0.060363
3	0.0	0.0	0.5000	0.0	0.0	0.0	0.0
4	0.0	0.0	0.7500	-0.678054	0.078763	-0.049384	-0.087266
4	0.0	0.0	0.7500	0.0	0.0	0.0	0.0
5	0.0	0.0	0.9167	-0.430516	0.427825	0.0	0.0
5	0.0	0.0	0.9167	0.0	0.0	0.0	0.0

BODY ELEM.	Y	Z	XOL	MODIFIED LIFT COEFFICIENTS, Z- AND Y-DIRECTIONS			
1	0.0	0.0	0.0833	0.361449	0.043729	0.0	0.0
2	0.0	0.0	0.2500	-0.630596	-0.051170	0.0	0.0
3	0.0	0.0	0.5000	1.215433	8.[illegible]40580	0.0	0.0
4	0.0	0.0	0.7500	-1.268764	0.836[illegible]63	0.0	0.0
5	0.0	0.0	0.9167	-0.812666	0.906133	0.0	0.0

BODY NO.	Y	Z	X-CENTER	TOTAL LIFT COEFF.		TOTAL MOMENT COEFF.	
1	0.0	0.0	3.0000	-0.291291	1.119119	0.943517	-2.342206
[illegible]	0.0	0.0	3.0000	0.0	0.0	0.0	0.0

BODY ELEM.	Y	Z	XOL	LIFT COEFFICIENT		MOMENT COEFFICIENT	
6	2.0000	-0.5000	0.0500	-0.431625	2.276594	0.000196	0.000299
6	2.0000	-0.5000	0.0500	0.443492	0.609356	-0.003399	-0.005177
7	2.0000	-0.5000	0.1500	-1.372394	7.491276	0.003208	0.006145
7	2.0000	-0.5000	0.1500	0.369820	1.849993	-0.003075	-0.005890
8	2.0000	-0.5000	0.3500	-1.836075	-0.749341	-0.003833	0.010936
8	2.0000	-0.5000	0.3500	-0.099934	0.205834	0.024563	0.027676
9	2.0000	-0.5000	0.7500	-0.594350	-2.152855	0.096763	-0.118245
9	2.0000	-0.5000	0.7500	0.008805	-0.474888	-0.017719	0.094607

BODY ELEM.	Y	Z	XOL	MODIFIED LIFT COEFFICIENTS, Z- AND Y-DIRECTIONS			
6	2.0000	-0.5000	0.0500	0.451001	2.988956	0.572218	0.627555
7	2.0000	-0.5000	0.1500	-1.039745	4.824802	0.344584	1.192759
8	2.0000	-0.5000	0.3500	-3.490913	-0.419818	-0.217383	0.532570
9	2.0000	-0.5000	0.7500	-1.962565	-5.778206	-0.053222	-1.257548

BODY NO.	Y	Z	X-CENTER	TOTAL LIFT COEFF.		TOTAL MOMENT COEFF.	
2	2.0000	-0.5000	3.2500	-0.401718	-0.126736	0.297814	0.481882
2	2.0000	-0.5000	3.2500	0.021779	0.027438	-0.000930	0.096011

CZ =	-1.253331	5.245323	CY =	0.0	0.0
CM =	1.816684	-2.882576	CN =	0.0	0.0
CSL=	0.0	0.0			

MODE NO 1

PANEL NO	STRIP NO	BOX NO	XOC	X	Y	Z	PRESSURES	
1	1	1	0.12500	2.35938	1.14952	0.50000	1.966562	4.242338
1	1	2	0.62500	3.29688	1.14952	0.50000	-1.103708	4.623168
1	2	3	0.12500	2.57813	1.71651	0.50000	2.693086	4.813315
1	2	4	0.62500	3.39063	1.71651	0.50000	-1.745509	4.937862
2	3	5	0.12500	2.79688	2.25000	0.62500	1.858071	4.368785
2	3	6	0.62500	3.48438	2.25000	0.62500	-1.807866	3.042441
2	4	7	0.12500	3.01563	2.75000	0.87500	1.684871	4.314862
2	4	8	0.62500	3.57813	2.75000	0.87500	-1.492552	1.997920
3	5	9	0.12500	2.68750	2.00000	0.25000	1.359550	1.076156
3	5	10	0.62500	3.43750	2.00000	0.25000	-0.076321	1.633911

THE 23 MODIFIED H-ELEMENTS FOR MODE NO. 1

1.59464	1.59464	1.38202	1.38202	0.99851	0.99851
0.81696	0.81696	0.0	0.0	1.50000	1.50000
1.50000	1.50000	1.50000	3.00000	3.00000	3.00000
3.00000	0.0	0.0	0.0	0.0	

THE 9 MODIFIED DH/DX-ELEMENTS FOR MODE NO. 1

0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0			

DEFLECTION MODE	PRESSURE MODE	GENERALIZED FORCES PANELS ONLY		TOTAL	
1	1	0.30132967E 00	0.41844883E 01	-0.24935694E 01	0.53777828E 01
1	2	0.52773800E 01	0.90802803E 01	0.96658516E 00	0.14132598E 02
1	3	-0.14895672E 00	0.43172178E 01	-0.12436838E 01	0.51828642E 01

MODE NO 2

PANEL NO	STRIP NO	BOX NO	XOC	X	Y	Z	PRESSURES	
1	1	1	0.12500	2.35938	1.14952	0.50000	8.323404	5.546045
1	1	2	0.62500	3.29688	1.14952	0.50000	3.644841	12.989089
1	2	3	0.12500	2.57813	1.71651	0.50000	10.454984	6.221313
1	2	4	0.62500	3.39063	1.71651	0.50000	2.416474	14.484083
2	3	5	0.12500	2.79688	2.25000	0.62500	8.435404	6.931283
2	3	6	0.62500	3.48438	2.25000	0.62500	-0.347866	10.142653
2	4	7	0.12500	3.01563	2.75000	0.87500	7.990103	7.380054
2	4	8	0.62500	3.57813	2.75000	0.87500	-0.996293	7.247329
3	5	9	0.12500	2.68750	2.00000	0.25000	3.764908	0.340826
3	5	10	0.62500	3.43750	2.00000	0.25000	2.020964	3.944560

THE 23 MODIFIED H-ELEMENTS FOR MODE NO. 2

2.50824	3.50489	2.37535	3.12395	1.86180	2.31945
1.64243	1.94279	0.0	0.0	0.50000	1.50000
3.00000	4.50000	5.50000	4.25000	4.75000	5.75000
7.75000	0.0	0.0	0.0	0.0	

THE 9 MODIFIED DH/DX-ELEMENTS FOR MODE NO. 2

1.00000	1.00000	1.00000	1.00000	1.00000	2.00000
2.00000	2.00000	2.00000			

DEFLECTION MODE	PRESSURE MODE	GENERALIZED FORCES PANELS ONLY		TOTAL	
2	1	0.11803073E 00	0.83202486E 01	-0.56997927E 01	0.80402927E 01
2	2	0.95952950E 01	0.18972519E 02	-0.29987698E 01	0.23680984E 02
2	3	-0.85063422E 00	0.87069788E 01	-0.31874332E 01	0.96464167E 01

MODE NO 3

PANEL NO	STRIP NO	BOX NO	XOC	X	Y	Z	PRESSURES	
1	1	1	0.12500	2.35938	1.14952	0.50000	1.774650	2.702881
1	1	2	0.62500	3.29688	1.14952	0.50000	-0.958163	4.093591
1	2	3	0.12500	2.57813	1.71651	0.50000	2.179616	4.175091
1	2	4	0.62500	3.39063	1.71651	0.50000	-2.112709	4.741535
2	3	5	0.12500	2.79688	2.25000	0.62500	1.698914	6.286385
2	3	6	0.62500	3.48438	2.25000	0.62500	-3.086831	3.671605
2	4	7	0.12500	3.01563	2.75000	0.87500	1.331918	7.792065
2	4	8	0.62500	3.57813	2.75000	0.87500	-3.092846	2.418398
3	5	9	0.12500	2.68750	2.00000	0.25000	0.718314	-0.043016
3	5	10	0.62500	3.43750	2.00000	0.25000	0.192435	0.883634

THE 23 MODIFIED H-ELEMENTS FOR MODE NO. 3

1.37274	1.37274	1.97330	1.97330	1.84577	1.84577
2.03752	2.03752	-0.09375	-0.09375	0.0	0.0
0.0	0.0	0.0	4.99999	4.99999	4.99999
4.99999	1.00000	1.00000	1.00000	1.00000	

THE 13 MODIFIED DH/DX-ELEMENTS FOR MODE NO. 3

0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0
0.0					

DEFLECTION MODE	PRESSURE MODE	GENERALIZED FORCES PANELS ONLY		TOTAL	
3	1	0.35609788E 00	0.60103865E 01	-0.52197278E 00	0.59655657E 01
3	2	0.73399734E 01	0.13251885E 02	0.54930477E 01	0.13903881E 02
3	3	-0.51838241E 00	0.68781071E 01	-0.18435221E 01	0.64739475E 01

*****

EXECUTION TIME(MINUTES)
CPU 0.0410
I/O 0.0377
TOTAL 0.0788

3.0 BLANK AND LABELED COMMON BLOCKS

3.1 Blank Common Block

ITEM NO.	SYMBOL	MNEMONICS	DESCRIPTION	
1		NP	No. of panels	Input 9
2		NTP	Total no. of boxes on all panels	
3		NB	No. of bodies	Input 10
4		NBZ	No. of bodies with z-orientation	
5		NBY	No. of bodies with y-orientation	
6		NTZ	Total no. of body elements with z-orientation	
7		NTY	Total no. of body elements with y-orientation	
8		FMACH	Mach Number	Input Items 2
9	A	REFA	Reference area	3
10	S	REFS	Reference span	4
11	$\bar{c}$	REFC	Reference chord	5
12	δ	ND	Symmetry flag for y = 0 plane	7
13	ε	NE	Symmetry flag for z = 0 plane	8
14		NBEA(10,2)	1. No. of interference elements (NBE) for all bodies 2. Z-Y flag (NZY) for all bodies	
15		NSBEA(10)	No. of slender body elements (NSBE) for all bodies	

ITEM NO.	SYMBOL	MNEMONICS	DESCRIPTION	
15		NSBEA(10)	No. of slender body elements (NSBE) for all bodies	
16	nc	NCARAY(100)	No. of chordwise boxes for all panels	
17	ns	NSARAY(100)	No. of spanwise strips for all panels	
18	nba	NBARAY(100)	$nba_i = \sum_{j=1}^{i} nc_j ns_j$, where $i = 1, NP$	
19	z_c	ZB(10)	z-coordinate of body center	
20	y_c	YB(10)	y-coordinate of body center	
21	$x_{L.E.}$	XLE(10)	x-coordinate of leading edge of body	Eq. (5.2.1-17)
22	$x_{T.E.}$	XTE(10)	x-coordinate of trailing edge of body	Eq. (5.2.1-18)
23	AR	ARB(10)	Cross sectional aspect ratio of body	
24	a	AVR(10)	Average characteristic half-width of body	Eq. (5.2.1-22)
25	a_o	AO(200)	Local body half-width (y-direction; radius for circle) Slender body arrays	
26	a_o	AOP(200)	x-derivative of body half-width	Eq. (5.2.1-23)
27	ξS1	XIS1(200)	x-coordinate of slender body element leading edge	Eq. (5.2.1-20)
28	ξS2	XIS2(200)	x-coordinate of slender body element trailing edge	Eq. (5.2.1-21)

ITEM NO.	SYMBOL	MNEMONICS	DESCRIPTION	
29		XIJ(200)	x-coordinate of leading edge of strips (on all panels)	
30	c	CS(210)	chordlength of strips	
31	y_{Ps},η_{Ps}	YS(210)	y-coordinate of strip center-lines and body center-lines	Eq. (5.2.1-9)
32	z_{Ps},ζ_{Ps}	ZS(210)	z-coordinate of strip center-lines and body center-lines	Eq. (5.2.1-10)
33	e_s	EE(200)	Half-width of strips	Eq. (5.2.1-11)
34	$\sin\gamma_s$	SG(200)	sine of the dihedral	Eq. (5.2.1-14)
35	$\cos\gamma_s$	CG(200)	cosine angle of strip	Eq. (5.2.1-15)
36	ξ_P, x_I	X(500)	3/4 chord x-coordinates of all boxes (receiving points) and interference body section midpoints	Eq. (5.2.1-1 and -2)
37	ξ_P	XIC(500)	1/4 chord x-coordinates of all boxes (sending points)	
38	k_r	KR	Reduced frequency	
39	XM	XM	Moment axis	Input Item 6
40	$\Delta x_P,\Delta x_I$	DELX(500)	Average chord-lengths of all boxes, and interference body element lengths	Eq. (5.2.1-7 and -8)
41	$\tan\lambda$	XLAM(500)	Tangent of the sweep angle of the 1/4-chord line (bound vortex) of all boxes	
42		RIA(100)	Radii of all body interference elements	

ITEM NO.	SYMBOL	MNEMONICS	DESCRIPTION
43		TH1A(100)	θ_1's for all bodies
44		TH2A(100)	θ_2's for all bodies
45		NFL(10)	Number of body sections with θ_1-distribution - for all bodies (max is 3 per body)
46		IFLA(30,2)	1. Sequence number of the first body element with θ_1 distribution 2. Sequence number of the last body element with θ_1-distribution - for all bodies
47		NT12(10,2)	1. Number of θ_1's | all bodies 2. Number of θ_2's |
48		NAS(100)	Number of associated bodies for all panels
49		NASB(200)	Associated bodies - all panels
50		YIN(100)	y-coordinate of inboard edge of panel
51		ZIN(100)	z-coordinate of inboard edge of panel - all panels
52	$\Delta\eta$	DETA(50)	y-shift elements — all panels and all bodies
53	$\Delta\zeta$	DZET(50)	z-shift elements — all panels and all bodies
54		NOBODY(50)	Body number — for the $\Delta\eta$, $\Delta\zeta$ pairs
55		NPANEL(50)	Panel number — for the $\Delta\eta$, $\Delta\zeta$ pairs
56		NTO	$NTP + \sum_{i=1}^{NB} NBE_i$, where NTP = total number of boxes on all panels and NBE_i = number of interference elements on body i

ITEM NO.	SYMBOL	MNEMONICS	DESCRIPTION
57		NTZS	Total no. of slender body elements with z-orientation
58		NTYS	Total no. of slender body elements with y-orientation

Subroutines using the Blank Common Block are:
MAIN, DATA, GEND, SUBP, SUBB, SB, BFM, AERO and GENF.

3.2 Labeled Common Blocks

Common DLM

ITEM NO.	SYMBOL	MNEMONICS	DESCRIPTION	
1	$K_1^{(s)}$	K10	Planar part of	the steady contribution to the kernel
2	$K_2^{(s)}$	K20	Nonplanar part of	
3	$Re\bar{K}_1$	K1RT1	$Re(\Delta K_1)T_1$	Unsteady part of modified kernel
4	$Im\bar{K}_1$	K1IT1	$Im(\Delta K_1)T_1$	
5	$Re\bar{K}_2$	K2RT2P	$Re(\Delta K_2)T_2*$	
6	$Im\bar{K}_2$	K2IT2P	$Im(\Delta K_2)(T_2*$	
7		K10T1	$K_1^{(s)}\ T_1$	
8		K20T2P	$K_2^{(s)}\ T_2^*$	

Subroutines using Common DLM are:

FLLD, INCRO and TKER.

Common KDS

ITEM NO.	SYMBOL	MNEMONICS	DESCRIPTION
1		IND	Flag, 0 or 1. - If 0, the total kernel is computed; if 1, the incremental part only.
2	$Re(\tilde{K}_1)$	KD1R	Real part of $\tilde{K}_1$
3	$Im(\tilde{K}_1)$	KD1I	Imaginary part of $\tilde{K}_1$
4	$Re(\tilde{K}_2)$	KD2R	Real part of $\tilde{K}_2$
5	$Im(\tilde{K}_2)$	KD2I	Imaginary part of $\tilde{K}_2$ - $\tilde{K}_1$ and $\tilde{K}_2$ are defined in Sec. 5.5.1.4 Subroutine FLLD.

Subroutines using Common KDS are:
FLLD, INCRO AND TKER.

4.0 LOGICAL TAPE UNITS

4.1 Tape Numbers and Symbols

TAPE NUMBER SYMBOLS	PRESENT ASSIGNMENT	USER SUBROUTINES
NTAPE(1),	1	GEND, SOLVIT
ITP1, NM, NDZ		MAIN
NTAPE(2),	2	GEND, SOLVIT
ITP2, NO, MTAPE, NDY		MAIN
NTAPE(3),	3	AERO, BFM, GEND, GENF,
ITP3, NW, NPTAP, NDW		SB, SOLVIT, WANDWT, MAIN, RWREC
NTAPE(4),	4	MAIN, GEND, SB, WANDWT
ITP4,NW5,NWT		
NTAPE(8),	8	MAIN, GEND, AERO, GENF
ITAPE, NEWBFM		
NTAPE(9),	9	BFM, SB, MAIN
NPSTAP,NW2		
NTAPE(10),	10	MAIN, SB, SOLVIT
NI, NW3		
NTAPE(11),	11	AERO, BFM, SB, MAIN, BFSMAT
NBFM,NW4		

In addition to the above tapes, units #5 and 6 are used throughout the program as the standard 'card-read' and 'print' units. The following table explains the use of the logical tape units within the user subroutines in the order of the program flow.

4.2 Table of Input Output Units

SUBROUTINE NAME	TAPE NUMBER				
	INPUT	CONTENTS	OUTPUT	CONTENTS	SCRATCH
GEND			ITAPE	[DT]	ITP1 ITP2 ITP3 ITP4
SB			NW1=NDW NW2 = NPSTAP	{ΔW} $\left\{\Delta C_p^{\,body}\right\}$	NW3 NW4 NW5
WANDWT	NDW	{ΔW}	NWT	{WT} ={W}+{ΔW}	
MAIN	ITAPE NWT	[DT] {WT}	NI	Augmented Matrix [DT ¦ WT]	
SOLVIT	NI	[DT ¦ WT]	NW	Solutions {P}	NM NO
BFSMAT			NBFM	BFS**	
MAIN	NBFM NW	BFS {P}	NEWBFM	BFM*	
BFM	NPSTAP NW	$\{\Delta C_p^{\,body}\}$ {P}	NBFM	BFM	
AERO	NW NBFM	{P} BFM	NEWBFM	BFM*	
GENF	NW NBFM	{P} BFM			

* BFM represents all the body forces and moments for all modes.

** BFS represents the body forces FZ and FY described in Sec. 5.8.1.

5.0 DESCRIPTION OF SUBROUTINES

5.1 Segment 1

5.1.1 MAIN

Functional Description

The MAIN part of program N5KA reads the header card, the reference variables, control and print flags, and the array of reduced frequencies for the case. It calls subroutine DATA which computes the basic data arrays - these are saved in the blank common block. The other 10 major subroutines are called from MAIN in an overall frequency do-loop for each element of the frequency array.

Input Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
FREQ(10)	k_r	IN	Card	Reduced frequency array
HDR(15)		IN	Card	Header information, one card
IA(2, 150,3)		IN	ARG of RDMODE	See Section 5.4.1
NA(3)			ARG of RDMODE	
DT(500)	[DT]	IN	I/O (ITAPE)	One row of the [DT] matrix generated in subroutine GEND
RHS(100)		OUT		One row of right-hand-sides for one unknown (panel box, or interference body element) and all modes
WORK (10000)	[WT]	IN	I/O (NWT)	Complete mode matrix for all unknowns and modes generated in subroutine WANDWT
RA (2, 10000)		OUT		A two-dimensional 'real' work array equivalenced with the complex work array WORK (10000) used in subroutine SOLVIT
NTAPE(20)		OUT	Data initialization	Variable name of the logical tape unit array
IERROR		OUT	MAIN	Error flag initialized to 0
NWORK		OUT	MAIN	Work array size

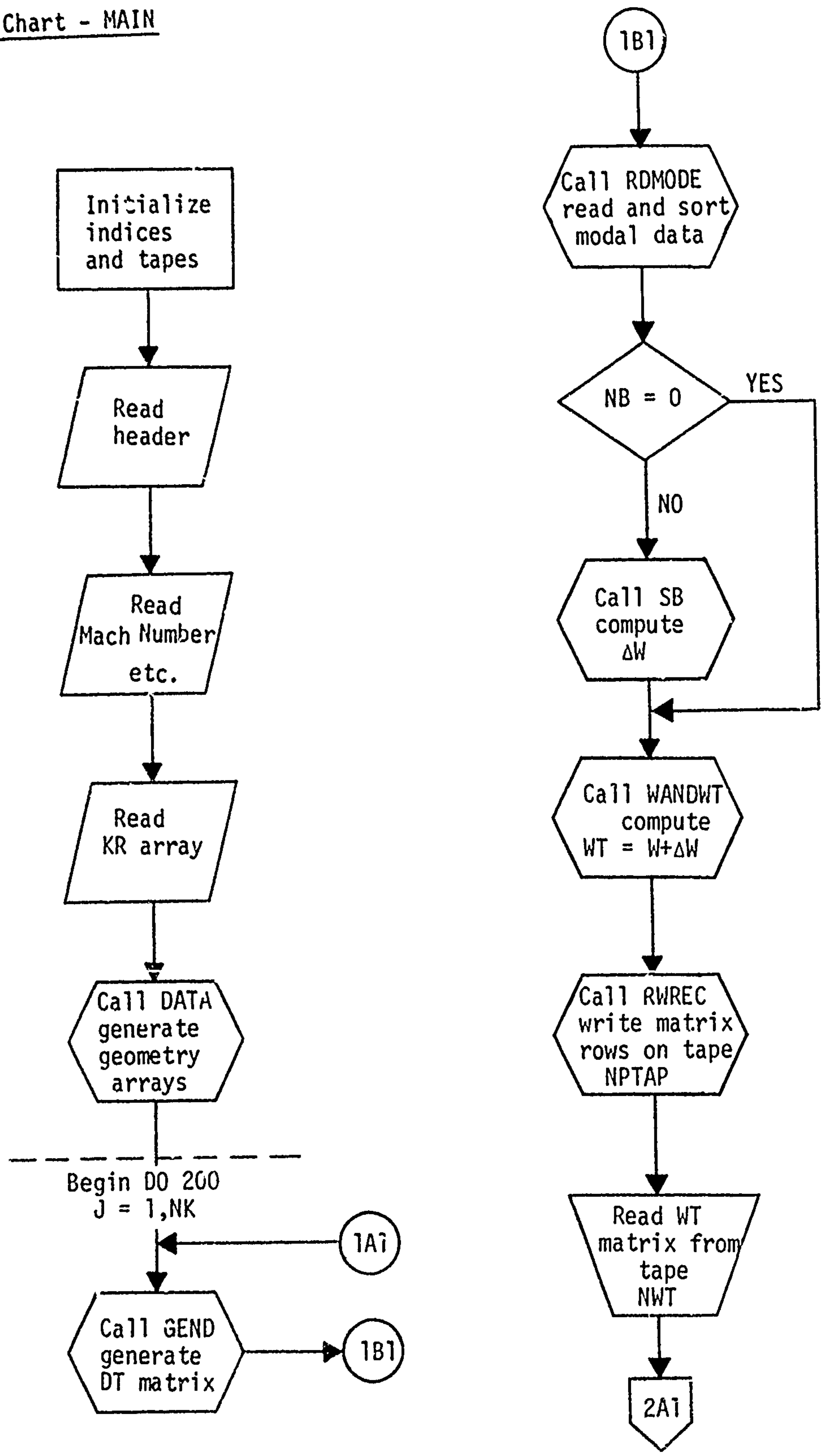
MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
NDZ				
NDY				
MTAPE				
ITAPE				
NPTAP				
NIN				
NOUT				
NPSTAP		IN	NTAPE (20)	See the argument list description of the user subroutines
NM				
NO				
NDW				
NWT				
NI				
NBFM				
NW				
FMACH	M			Mach Number
REFA	A			Reference area
REFS	s			Reference semispan
REFC	$\bar{c}$			Reference chord
XM				Moment axis
ND	δ			Symmetry flag
NE	ε			Second symmetry flag
NP				No. of panels
NB		IN	Card	No. of bodies
NK				No. of reduced frequencies
N1				
N2				
N3				
N4				Print and control flags
IBFS				Option flag for body force calculation method; 0 or 1
NEWBFM				Number of logical tape unit containing body forces

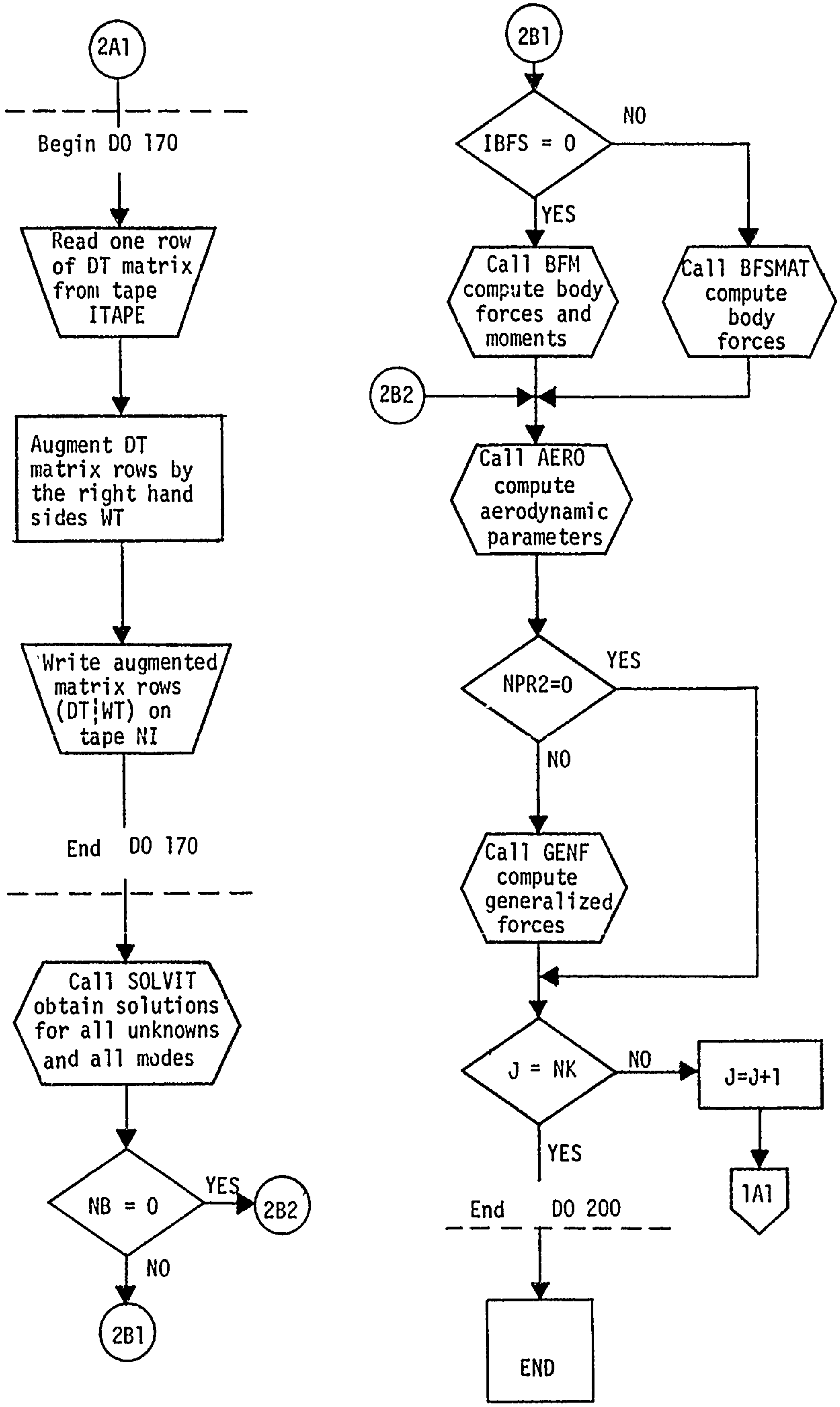
MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
BETA	β			$\beta = \sqrt{1 - M^2}$
NPR1,N1				Print flag for SØLVIT
NPR2,N2				Control flag for GENF
NPRINT,N3				Print flag for GEND
NTØT				Total number of unknowns
NTIBE				Number of interference body elements
NTSBE				Number of slender body elements
N		OUT	MAIN	Number of panel boxes plus number of interference body elements
NSTRIP				Total number of strips on all panels
ID,NTOT				See NTOT above
KD				Total dimension of the real work-array RA
NR,NTOT				See NTOT above
KR	$\hat{r}_j$			j-th element of reduced frequency array
MD,NMT		IN	Arg list of RDMODE	Total number of modes
NMODE,MD		OUT		Total number of modes
NIM				
NFC				See Section 5.5.8.1
NWKR				
JX1				Index of first-(JX1) and last (JX2)
JX2		IN	MAIN	elements in the modal matrix array when read from logical unit number NWT one column at a time
JM				Indices used in the do loop transposing
JJ				the modal matrix array
JX				

Subroutines Called DATA, GEND, RDMODE, SB, WANDWT, RWREC, SOLVIT, BFM, BFSMAT, AERO and GENF

Common Blocks Blank Common Block

Flow Chart - MAIN




2A1
Begin DO 170
Read one row of DT matrix from tape ITAPE
Augment DT matrix rows by the right hand sides WT
Write augmented matrix rows (DT¦WT) on tape NI
End DO 170
Call SOLVIT obtain solutions for all unknowns and all modes
NB = 0
YES
2B2
NO
2B1
2B1
IBFS = 0
NO
YES
Call BFM compute body forces and moments
Call BFSMAT compute body forces
2B2
Call AERO compute aerodynamic parameters
NPR2=0
YES
NO
Call GENF compute generalized forces
J = NK
NO
J=J+1
1A1
YES
End DO 200
END

5.1.2 Subroutine ATAN3 (Y, X, T)

Functional Description

Subroutine ATAN3 evaluates T = atan (Y,X) by considering the signs of both Y and X, and thereby providing a result, T, that lies in the proper quadrant. The resulting angle, T, is returned to the calling program in radians.

Calling Subroutine DATA

5.1.3 SUBROUTINE DZY (X, Y, Z, SGR, CGR, XI1, XI2, ETA, ZETA, AR, AO, KR, CBAR, BETA, FMACH, IDZDY, DZDYR, DZDYI)

Functional Description

Subroutine DYZ calculates the effect of a finite length doublet, a multi finite length doublet, or a trapezoidal vortex, of unit strength, at a field point. Each slender body is composed of a series of segments or elements. This subroutine calculates the effect of such elements on field points (lifting surface boxes, etc.). If the cross sectional shape is circular then a doublet is used. If AR > 1, two doublets are used. If AR < 1 a trapezoidal vortex is used. This subroutine is used for bodies oriented in both the z- and y-directions.

Input Output Variables

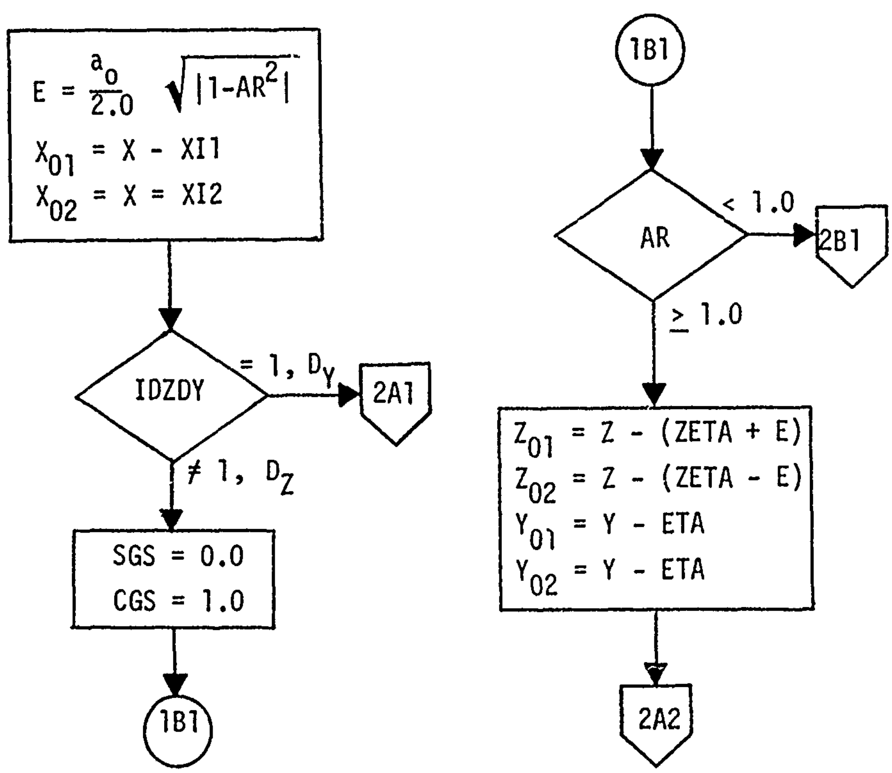
MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
X	X			x-coordinate of the receiving point
Y	Y			y-coordinate of the receiving point
Z	Z			z-coordinate of the receiving point
AO	a_0			Radius of sending body
AR	$\AE$			Aspect ratio of sending body
KR	k_r	IN	ARG	Reduced frequency
CGR	$\cos\gamma_r$			Cosine of receiving point dihedral angle
ETA	η			y-coordinate of the sending strip
SGR	$\sin\gamma_r$			Sine of receiving point dihedral angle
XI1	ξ_1			Leading edge of sending element
XI2	ξ_2			Trailing edge of sending element

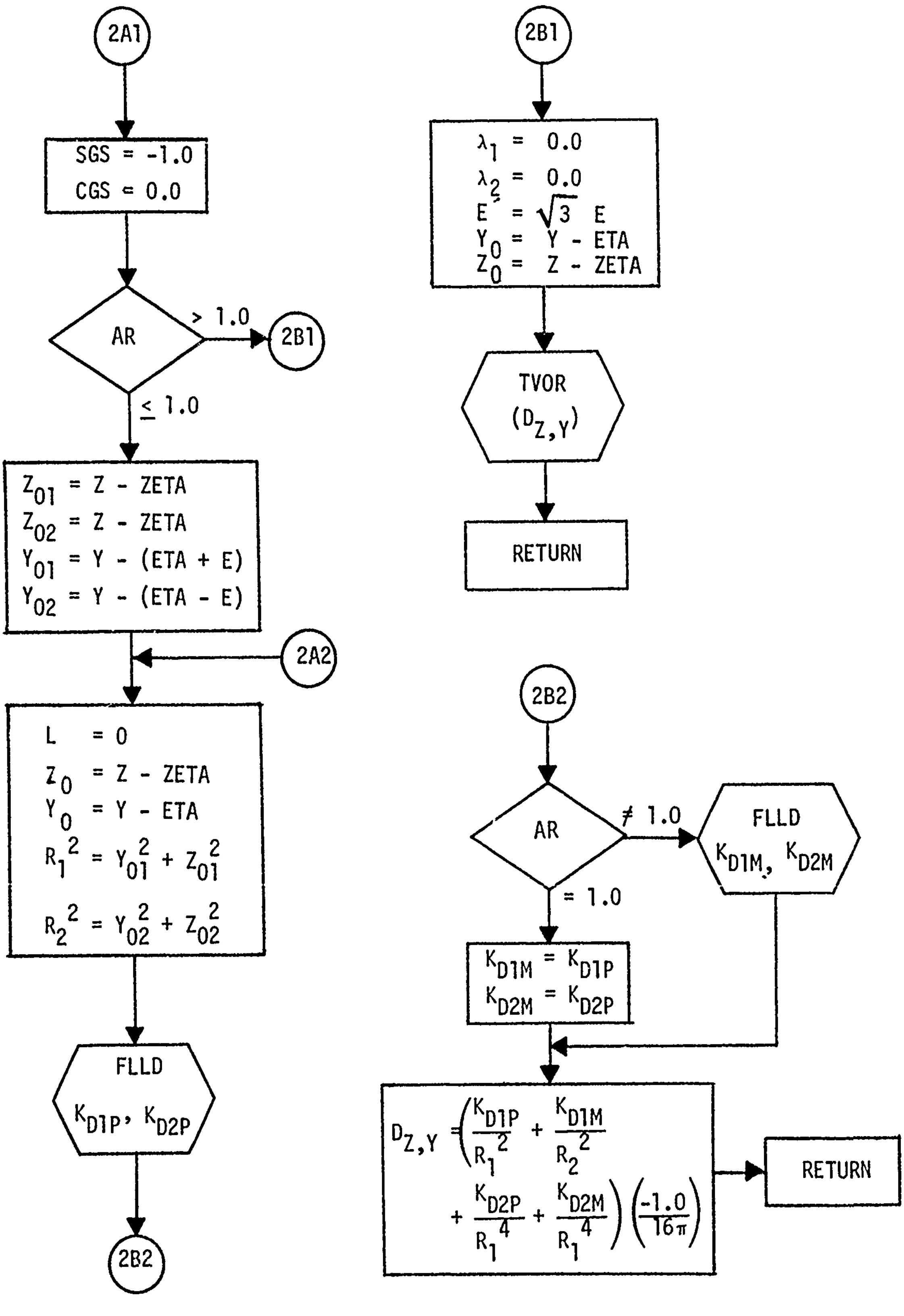
MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
BETA CBAR ZETA	β $\bar{c}$ ζ	IN	ARG	$\sqrt{1 - M^2}$ Reference chord length z-coordinate of the sending strip
DZDYI	$Im(D_z)$ $Im(D_y)$	OUT	ARG	Imaginary part of D_z or D_y
DZDYR	$Re(D_z)$ $Re(D_y)$	OUT	ARG	Real part of D_z or D_y
FMACH IDZDY	M	IN	ARG	MACH number FLag indicating whether D_z or D_y is to be calculated

Calling Subroutines   ROWDYZ

Called Subroutines   FLLD, TVOR

Flow Chart - Subroutine DZY




2A1
SGS = -1.0
CGS = 0.0
AR
> 1.0
2B1
≤ 1.0
Z_{01} = Z - ZETA
Z_{02} = Z - ZETA
Y_{01} = Y - (ETA + E)
Y_{02} = Y - (ETA - E)
2A2
L = 0
Z_0 = Z - ZETA
Y_0 = Y - ETA
$R_1^2 = Y_{01}^2 + Z_{01}^2$
$R_2^2 = Y_{02}^2 + Z_{02}^2$
FLLD
K_{D1P}, K_{D2P}
2B2
2B1
λ_1 = 0.0
λ_2 = 0.0
$E = \sqrt{3}\ E$
Y_0 = Y - ETA
Z_0 = Z - ZETA
TVOR
$(D_{Z,Y})$
RETURN
2B2
AR
≠ 1.0
FLLD
K_{D1M}, K_{D2M}
= 1.0
$K_{D1M} = K_{D1P}$
$K_{D2M} = K_{D2P}$
$D_{Z,Y} = \left(\frac{K_{D1P}}{R_1^2} + \frac{K_{D1M}}{R_2^2} + \frac{K_{D2P}}{R_1^4} + \frac{K_{D2M}}{R_1^4}\right)\left(\frac{-1.0}{16\pi}\right)$
RETURN

5.1.4 SUBROUTINE FLLD (XO1 , XO2, YO,ZO, SGR, CGR, SGS, CGS, KR, CBAR, FMACH, E, L, KD1R, KD1I, KD2R, KD2I)

Functional Description

This subroutine calculates the velocity normal to a surface of dihedral, γ_r, due to a finite length line doublet.

Input Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
E	e	IN	ARG	Semi width of singularity
L		IN	ARG	Option flag for use in subroutine TKER
KR	k_r	IN	ARG	Reduced frequency
YO	$y-\eta$	IN	ARG	Difference in lateral coordinates of receiving and sending points respectively
ZO	$z-\zeta$	IN	ARG	Difference in vertical coordinates of receiving and sending points respectively
CGR	$\cos\gamma_r$	IN	ARG	
CGS	$\cos\gamma_s$	IN	ARG	
SGR	$\sin\gamma_r$	IN	ARG	
SGS	$\sin\gamma_s$	IN	ARG	
XO1	$x-\xi_1$	IN	ARG	Difference in longitudinal coordinates of receiving and leading edge of sending points respectively
XO2	$x-\xi_2$	IN	ARG	Same as above except ξ_2 is the trailing edge
CBAR	$\bar{c}$	IN	ARG	Reference chord
KD1I	$Im(K_{d1})$	OUT	ARG	See Equation (5.1.4-1)
KD1R	$Re(K_{d1})$	OUT	ARG	See Equation (5.1.4-1)
KD2I	$Im(K_{d2})$	OUT	ARG	See Equation (5.1.4-2)
KD2R	$Re(K_{d2})$	OUT	ARG	See Equation (5.1.4-2)
FMACH	M	IN	ARG	Mach number
E2	e^2	OUT	DLM	

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
IND	L	OUT	KDS	
KK1I KK1R KK2I KK2R	$I_m(KK1)$ $R_e(KK1)$ $I_m(KK2)$ $R_e(KK2)$	IN	KDS	See Section 3.2
K1OT1 KZOT2P		IN	DLM	See Section 3.2

Calling Subroutines TVOR, DZY

Called Subroutines and Common Blocks

TKER, KDS, DLM

Equations

$$K_{d1} = \tilde{K}_1(\xi_1)\ e^{ik_r\Delta\xi/\bar{c}} - \tilde{K}_1(\xi_2)\ e^{-ik_r\Delta\xi/\bar{c}} + K_{d1r}\ L \qquad (5.1.4\text{-}1)$$

$$K_{d2} = \tilde{K}_2(\xi_1)\ e^{ik_r\Delta\xi/\bar{c}} - \tilde{K}_2(\xi_2)\ e^{-ik_r\Delta\xi/\bar{c}} + K_{d2r}\ L \qquad (5.1.4\text{-}2)$$

where

$$\Delta\xi = \xi_2 - \xi_1$$

$$\tilde{K}(\xi) = \text{output from subroutine TKER}$$

$$K_{d1r} = -T_1\ (K_{10}(\xi_1) - K_{10}(\xi_2))$$

$$K_{d2r} = -T_2^*\ (K_{20}(\xi_1) - K_{20}(\xi_2))$$

T_1, T_2^*, K_{10}, K_{20} are output from TKER

L is an option flag for TKER and is either unity or zero.

5.1.5 SUBROUTINE FWMW (ND, NE, SGS, CGS, IRB, AO, ARB, XBLE, XBTE, YB, ZB, XS, YS, ZS, NAS, NASB, KR, BETA2, CBAR, FWZ, FWY, MWZ, MWY, IF1, IPRNT, IBFS)

Functional Description

Given a unit pressure doublet this subroutine calculates the effect of this doublet plus any contributions due to images, symmetry plane and ground effect on a given body.

Input Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
ND	δ			Symmetry flag.
NE	ε			Ground effects flag.
SGS	$\sin\gamma_s$			Sine of sending point dihedral angle
CGS	$\cos\gamma_s$			Cosine of sending point dihedral angle
IRB				Number of the receiving body
AO	a_o			Radius of the body
ARB	$A\!\!R$			Array of ratios of body axes
XBLE				Leading edge location of slender body element
XBTE				Trailing edge location of slender body element
YB	Y_B	IN	ARG	Array containing the y-coordinates of the bodies
ZB	Z_B			Array containing the z-coordinates of the bodies
XS				1/4-chord x-coordinate of slender body element
YS				y-coordinate of sending point
ZS				z-coordinate of sending point
NAS				Number of associated bodies
NASB				Array containing the associated body numbers
KR	k_r			Reduced frequency
BETA2	β^2			$1-M^2$
CBAR	$\bar{c}$			Reference chord length
FWZ	F_{wz}			Force in z direction
FWY	F_{wy}	OUT	ARG	Force in y direction

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
MWZ MWY	M_{wz} M_{wy}	OUT	ARG	Moment in z direction Moment in y direction
IF1		IN	ARG	Flag indicating the orientation of the receiving body
IPRINT				Print flag
IBFS		IN/OUT	ARG	Option flag; 0 to select subroutine FMZY, 1 to select subroutine FZY2

Calling Subroutines BFM, BFSMAT

Called Subroutines and Common Blocks FMZY, SUBI, FZY2

Equations

$$
\begin{aligned}
FWZ = {} & \cos\gamma_s F_{zz} - \sin\gamma_s F_{zy} \\
& + \sum_{bI=range} [\tilde{\mu}_z F_{zz}{}^{I} + \tilde{\mu}_y F_{zy}{}^{I}] \\
& + \delta\left\{ \cos\gamma_s F_{zz}{}^{S} + \sin\gamma_s F_{zy}{}^{S} + \sum_{bI\ range} \left|\tilde{\mu}_z F_{zz}{}^{I(S)} + \tilde{\mu}_y F_{zy}{}^{I(S)}\right|\right\} \\
& + \varepsilon\left\{ \cos\gamma_s F_{zz}{}^{G} + \sin\gamma_s F_{zy}{}^{G} + \sum_{bI-range} \left|\tilde{\mu}_z F_{zz}{}^{I(G)} + \tilde{\mu}_y F_{zy}{}^{I(G)}\right|\right\} \\
& + \delta\varepsilon\left\{ \cos\gamma_s F_{zz}{}^{S(G)} + \sin\gamma_s F_{zy}{}^{S(G)} + \sum_{bI-range} \left|\tilde{\mu}_z F_{zz}{}^{I(S(G))} + \tilde{\mu}_y F_{zy}{}^{I(S(G))}\right|\right\}
\end{aligned}
$$

FWY = same as FWZ replacing F_{zz} with F_{yz} and replacing F_{zy} with F_{yy}

MWZ - same as FWZ replacing F_{zz} with M_{zz} and F_{zy} with M_{zy}

MWY - same as FWZ replacing F_{zz} with M_{yz} and F_{zy} with M_{yy}

The superscripts on F correspond to

I - Image points
S - Symmetry points
G - Ground effect points

bI range refers to the bodies associated with the lifting surface sending element

5.1.6 SUBROUTINE IDF1 (EE, E2, ETA)1, ZETO1, ARE, AIM, BRE, BIM, CRE, CIM, R1SQX, XIIJR, XIIJI)

Functional Description

Subroutine IDF1 performs the integration of the planar parts of the incremental oscillatory kernels according to Equation (B.9)*. The result of the integration is the complex number (XIIJR, XIIJI) which is returned to subroutine INCRO via the argument list of subroutine IDF1.

Input Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
EE	e	IN	ARG	Semi-width of sending element
E2	e^2	IN	ARG	
ETA01	$\bar{y}$	IN	ARG	$y_0 \cos\gamma_r + z_0 \sin\gamma_r$
ZET01	$\bar{z}$	IN	ARG	$z_0 \cos\gamma_r - y_0 \sin\gamma_r$
ARE	$Re(A_1)$	IN	ARG	Coefficients of the parabola for the planar part of kernel integration - See Eq's (B.3) through (B.5)
AIM	$Im(A_1)$	IN	ARG	
BRE	$Re(B_1)$	IN	ARG	
BIM	$Im(B_1)$	IN	ARG	
CRE	$Re(C_1)$	IN	ARG	
CIM	$Im(C_1)$	IN	ARG	
R1SQX	$r_1^{\,2}$	IN	ARG	$\bar{y}^2 + \bar{z}^2$
XIIJR	$Re(D_{1rs})$	OUT	ARG	Real part of integral } planar contribution See Eq. (B.9)
XIIJI	$Im(D_{1rs})$	OUT	ARG	Imaginary part of integral } planar contribution See Eq. (B.9)
TRM2R		OUT	IDF1	Real part } of the second term inside the brackets
TRM2I		OUT	IDF1	Imaginary part } of the second term inside the brackets
TRM3R		OUT	IDF1	Real part } of the third term inside the brackets of Eq. (B.9)
TRM3I		OUT	IDF1	Imaginary part } of the third term inside the brackets of Eq. (B.9)

* Appendix B in Part I Vol. I of this report (Reference 2)

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
TESTO				A test value: $\dfrac{\bar{y}^2 + \bar{z}^2 - e^2}{2\,e\lvert\bar{z}\rvert}$
TEST				Alternate test value: $\dfrac{2\,e\,\bar{z}}{\bar{y}^2 + \bar{z}^2 - e^2}$
ARGT				The argument to the arctangent in Eq. (B.6b)
		OUT	IDF1	
S				The argument to the series in Eq. (B.8)
SER				The sum of the series in Eq. (B.8)
ALPHA				See Eq. (B.8)
FUNCT	F			See Eq. (B.6b) and Eq. (B.7)
TRM1R				Real part } of the first term inside the brackets of Eq. (B.9)
TRM1I				Imaginary part } of the first term inside the brackets of Eq. (B.9)

<u>Calling Subroutine</u> INCRO

<u>Equations</u> - See Appendix B in Part I Volume I of this report (Reference 2)

5.1.7 SUBROUTINE IDF2 (EE, E2, ETA01, ZET01, A2R, A2I, B2R, B2I, C2R, C2I, R1SQX, DIIJR, DIIJI)

Functional Description

Subroutine IDF2 performs the integration of the nonplanar parts of the incremental oscillatory kernels according to Equations (B.15) or (B.16)*. The result of the integration is the complex number (DIIJR, DIIJI) which is returned to subroutine INCRO via the argument list of subroutine IDF2.

Input Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
EE	e	IN	ARG	See Subroutine IDF1
E2	e^2	IN	ARG	See Subroutine IDF1
ETA01	$\bar{y}$	IN	ARG	See Subroutine IDF1
ZET01	$\bar{z}$	IN	ARG	See Subroutine IDF1
A2R	$Re(A_2)$	IN	ARG	Coefficients of the parabola for the nonplanar part of the kernel integration - See Eq's (B.12) - (B.14)
A2I	$Im(A_2)$	IN	ARG	Coefficients of the parabola for the nonplanar part of the kernel integration - See Eq's (B.12) - (B.14)
B2R	$Re(B_2)$	IN	ARG	Coefficients of the parabola for the nonplanar part of the kernel integration - See Eq's (B.12) - (B.14)
B2I	$Im(B_2)$	IN	ARG	Coefficients of the parabola for the nonplanar part of the kernel integration - See Eq's (B.12) - (B.14)
C2R	$Re(C_2)$	IN	ARG	Coefficients of the parabola for the nonplanar part of the kernel integration - See Eq's (B.12) - (B.14)
C2I	$Im(C_2)$	IN	ARG	Coefficients of the parabola for the nonplanar part of the kernel integration - See Eq's (B.12) - (B.14)
R1SQX	$r_1{}^2$	IN	ARG	$\bar{y}^2 + \bar{z}^2$
DIIJR	$Re(D_{2rs})$	OUT	IDF2	Real part of integral — Nonplanar contribution See Eq (B.15) or (B.16)
DIIJI	$Im(D_{2rs})$	OUT	IDF2	Imaginary part of integral — Nonplanar contribution See Eq (B.15) or (B.16)
TESTO		OUT	IDF2	Test value: $\left\lvert \dfrac{\bar{y}^2 + \bar{z}^2 - e^2}{2\, e\, \bar{z}} \right\rvert$
TRM2R		OUT	IDF2	Real part of the second term inside the brackets of Eq (B.15) or same of Eq (B.16)
TRM2I		OUT	IDF2	Imaginary part of the second term inside the brackets of Eq (B.15) or same of Eq (B.16)

* Appendix B in Part I Vol. I of this report (Reference 2)

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
TRM3R				Real part } of the third term inside the brackets
TRM3I				Imaginary part } of the third term inside the brackets
TEST				Alternate test value: $\dfrac{2\,e\,\bar{z}}{\bar{y}^2 + \bar{z}^2 - e^2}$
S				See Subroutine IDF1
SER		OUT	IDF2	See Subroutine IDF1
ALPHA	α			See Subroutine IDF1
FUNCT	F			See Subroutine IDF1
ARGT				See Subroutine IDF1
TRM1R				Real part } of the first term inside the brackets of Eq. (B.15) or same of Eq. (B.16)
TRM1I				Imaginary part } of the first term inside the brackets of Eq. (B.15) or same of Eq. (B.16)

Calling Subroutine INCRO

Equations - See Appendix B in Part I Vol. I of this report (Reference 2)

5.1.8 SUBROUTINE INCRO (AX, AY, AZ, AX1, AY1, AZ1, AX2, AY2, AZ2, SGR, CGR, SGS, CGS, KR, FL, BETA, SDELX, DELY, DELR, DELI, IO, IR, NBXS, NCPNB, LHS, NDBLE, IMG, NOBI, IMGS, USE1, USE2, USE3, USE4, XUSE1, XUSE2, XUSE3, XUSE4)

Functional Description

Subroutine INCRO prepares the arguments for the subroutines TKER, IDF1 and IDF2. It calls subroutine TKER which computes the incremental oscillatory part of the kernel K for each receiving-sending box combination at the three points of the bound vortex segment: at the center (K_c), at the inboard point (K_i) and at the outboard point (K_o). Since even a relatively small case requires many kernel computations (e.g. an unsteady 100-box all

panel case, in symmetry, requires 2x100x3x100 = 60,000 kernel values) extra programming effort was made to reduce the number of kernel computations. Neighboring strips have common kernels on the common boundary lines; this property is utilized in subroutine INCRO for all lifting surface strips as well as their images inside associated bodies, by saving the arrays of kernels for all common boundary lines and using these in the subsequent calculations.

After the triplet of kernels, (K_c, K_i, K_o) is obtained for one receiving-sending box combination, subroutine INCRO computes the coefficients of the parabolas for the numerical integrations, done in subroutines IDF1 and IDF2.

Input Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION	
AX	$\bar{x}$			$x - \xi$	where (x, y, z)
AY	$\bar{y}$			$y - \eta$	define the receiving
AZ	$\bar{z}$			$z - \zeta$	point, and ξ, η, ζ are
AX1				$x - \xi_1$	the 'center' sending
AY1		IN	ARG	$y - \eta_1$	point coordinates,
AZ1				$z - \zeta_1$	ξ_1, η_1, ζ_1 are the
AX2				$x - \xi_2$	'inboard' sending point
AY2				$y - \eta_2$	coordinates and
AZ2				$z - \zeta_2$	ξ_2, η_2, ζ_2 are the
					'outboard' sending
					point coordinates

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
SGR		IN	ARG	$\sin\gamma_r$ — where 'r' is the index of the receiving point and 's' is the index of the sending point
CGR		IN	ARG	$\cos\gamma_r$
SGS		IN	ARG	$\sin\gamma_s$
CGS		IN	ARG	$\cos\gamma_s$
KR	k_r	IN	ARG	Reduced frequency
FL	$\bar{c}/2$	IN	ARG	Reference semi-chord
BETA	β	IN	ARG	$\beta = \sqrt{1 - M^2}$
SDELX	Δx_s	IN	ARG	Average box length of sending point 's'
DELY	$2\ e_s$	IN	ARG	Box width of sending point 's'
DELR		OUT	ARG	Real part of the oscillatory contribution to the nonplanar downwash factor
DELI		OUT	ARG	Imaginary part of the oscillatory contribution to the nonplanar downwash factor
IO		IN	ARG	An index running from 1 through $NC^{(p)}$, where $NC^{(p)}$ is the no. of chordwise boxes in the sending panel p
IR		IN	ARG	Index of sending point
NBXS		IN	ARG	The no. of boxes on the panel in which the sending point lies + the total no. of boxes of the preceding panels
NCPNB		IN	ARG	The no. of boxes in the first strip of the panel in which the sending point lies + the total no. of boxes of the preceding panels

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
LHS		IN	ARG	Flag activated by δ (input) for the contribution of the first symmetry plane effect
NDBLE				Flag activated by ϵ (input) for the contribution of the second symmetry plane effect
IMG				Flag activated by the presence of image points inside associated bodies for sending panel
NOBI				Sequence number of the body in which the image of the sending point lies
IMGS				IMGS = IMG whenever kernels are saved for image sending points; IMGS = 0 otherwise
K10	$K_1^{(s)}$	IN	Labeled Common DLM	Planar part of the steady contribution to the kernel
K20	$K_2^{(s)}$			Nonplanar part of the steady contribution to the kernel
K1RT1 K1IT1 K2RT2P K2IT2P K10T1 K20T2P				See Sec. 3.2
E2	e^2	OUT		

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
IND		OUT	Labeled Common KDS	See Sec. 3.2
KD1R	$Re(Kd_1)$	IN		
KD1I	$Im(Kd_1)$			
KD2R	$Re(Kd_2)$			
KD2I	$Im(Kd_2)$			
USE1(50, 14)		IN	ARG	Utility arrays for use of the planar kernel values which are saved for future use
USE2(50, 14)				
USE3(50, 14)				
USE4(50, 14)				
XUSE1(50, 14)		IN	ARG	Utility arrays for use of the nonplanar kernel values which are saved for future use
XUSE2(50, 14)				
XUSE3(50, 14)				
XUSE4(50, 14)				
M	M	OUT	INCR'O	Mach Number
BR	$\bar{c}/2$			Reference semi-chord
EPS				$\epsilon = 0.00001$
PI	π			
XDELX				See Subroutine IDF1
XDELY				
EE				
E2				
COUNT				An internal flag to select logic of subroutine

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
XO YO ZO		OUT	INCRO	Differences of the x, y and z coordinates of the current receiving and sending points (see AX, AY, AZ, AX1, AY1, AZ1 and AX2, AY2, AZ2 above)
AT1 AT2	$\|T_1\|$ $\|T_2\|$	IN	Argument List of TKER	See Appendix A in Part I, Vol I of this report
DKRC DKIC XKRC XKIC		IN	INCRO	Planar kernel / Nonplanar kernel — for the center point of the bound vortex segment
DKRI KDJI XKRI XKII		IN	INCRO	Planar kernel / Nonplanar kernel — for the inboard point of the bound vortex segment
DKRO DKIO XKRO XKIO		IN	INCRO	Planar kernel / Nonplanar kernel — for the outboard point of the bound vortex segment
JO		IN	INCRO	Index for the selection of the proper utility array in which the kernels are saved
XMULT		IN	INCRO	$\Delta x_j/8\pi$
ETA01 ZET01 R1SQX ARE		OUT	INCRO	See Subroutine IDF1

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
AIM		OUT	INCRO	See Subroutine IDF1
BRE				
BIM				
CRE				
CIM				
XIIJR		IN	Argument List of IDF1	
XIIJI				
A2R		OUT	INCRO	See Subroutine IDF2
A2I				
B2R				
B2I				
C2R				
C2I				
DIIJR		IN	Argument List of IDF2	
DIIJI				
DELR	$Re(\Delta D_{rs})$	OUT	INCRO	$\Delta D_{rs} = D_{1rs} + D_{2rs}$
DELI	$Im(\Delta D_{rs})$			See Appendix B in Part I, Vol I of this report

<u>Calling Subroutine</u> SUBP

<u>Called Subroutines and Common Blocks</u>

Subroutines [illegible] IDF1 and IDF2, and the Labeled Common Blocks DLM and KDS.

<u>Equations</u> See Appendix B in Part I Vol. I of this report (Reference 2)

5.1.9 SUBROUTINE READD (D, N, NTAPE)

Functional Description

This routine reads the complex array D which is N complex words long from the I/O unit NTAPE.

Input Output Variables

MNEMONICS	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
D		OUT	ARG	Complex array read
N		IN	ARG	Number of complex words in array D
NTAPE		IN	ARG	Input unit number

Calling Subroutines OUTCOR, SB

5.1.10 SUBROUTINE RWREC (IFLAG, NTAP, A, NCWORD, NUMBR)

Functional Description

This routine is used to read and write the array A which is NCWORD complex words long on the I/O unit NTAP.

Input-Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
IFLAG		IN		Read-write flag = 0 write A = 1 read A = 2 write NUMBR, A = 3 read NUMBR, A
NTAP		IN	ARG	I/O unit number to use
A		IN/OUT		Complex array to be read or written
NCWORD		IN		Length of array A
NUMBR		IN/OUT		The number to precede array A if desired; zero otherwise

Calling Subroutines BFM, BFSMAT, MAIN, OUTCOR, SB

5.1.11 SUBROUTINE SNPDF (SL, CL, TL, SGS, CGS, SGR, CGR, XO, YO, ZO, EE, EIJ, BETA, CV)

Functional Description

Subroutine SNPDF computes the steady downwash factors for one receiving-sending point combination at a time according to Eq. (C.35b), Appendix C*. The result, DIJ, is returned to the calling subroutine via the argument list of subroutine SNPDF.

Input Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
SL				$\sin\lambda$ where λ is the
CL				$\cos\lambda$ sweep angle of the
TL				$\tan\lambda$ 1/4-chord line of the sending box, i.e. the sweep angle of the bound vortex
SGS		IN	ARG	
CGS				
SGR				See Subroutine INCRO
CGR				
XO	$\bar{x}$			AX
YO	$\bar{y}$			AY
ZO	$\bar{z}$			AZ
EE	e			
DIJ	$D_{rs}^{(s)}$	OUT	SNPDF	Steady contribution to the downwash factor See Eq. (C.35b)
BETA	β	IN	ARG	See Subroutine INCRO
CV	Δx_s			

* Appendix C in Part IVol. I of this report (Reference 2)

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
RIMAG	R_i			See Eq. (C.14)
ROMAG	R_o			See Eq. (C.16)
DB2	d_b^2			See Eq. (C.30)
VBY	V_{by}			
VBZ	V_{bz}	IN	SNPDF	
VIZ	V_{iy}			See Eq. (C.33b)
VIZ	V_{iz}			
VOY	V_{oy}			
VOZ	V_{oz}			
WW	w			See Eq. (C.34b)

<u>Calling Subroutines</u> SUBP and TVØR

<u>Equations</u> See Appendix C in Part I, Vol. I of this report (Reference 2)

5.1.12 <u>SUBROUTINE SUBI (DA, DZB, DYB, DAR, DETA, DZETA, DCGAM, DSGAM, DEE, DXI, TL, DETAI, DZETAI, DCGAMI, DSGAMI, DEEI, DTLAMI, DMUY, DMUZ, INFL, IQTFL</u>

<u>Functional Description</u>

Subroutine SUBI has a dual role: depending on the setting of the flag INFL, subroutine SUBI either computes the arguments necessary for the calculation of the influence coefficient-contribution of lifting surface image points inside associated bodies (INFL = 0), or it calculates the $\tilde{\mu}_y$, $\tilde{\mu}_z$ for one body element according to Eqs.(5.1.12-21 and -22) respectively (INFL = 1).

Input Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
DA	a	IN	ARG	Body data
DZB	z_c			
DYB	y_c			
DAR	AR			
DETA	η_s			Sending point data (on panels)
DZETA	ζ_s			
DCGAM	$\cos\gamma_s$			
DSGAM	$\sin\gamma_s$			
DEE	e_s			
DXI	ξ			
TL	$\tan\lambda$			
DETAI	ηI	OUT		y-coordinate of image point (Eq. 5.1.12-1 or -19)
DZETAI	ζI			z-coordinate of image point (Eq. 5.1.12-2 or -20)
DCGAMI	$\cos\gamma I$			cosine of the dihedral angle of the image of the bound vortex plane (Eq. 5.1.12-3 and -4)
DSGAMI	$\sin\gamma I$			sine of the dihedral angle of the image of the bound vortex plane (Eq. 5.1.12-3 and -4)
DEEI	eI			Semi-width of image of sending strip (Eq. 5.1.12-6)
DTLAMI	$\tan\lambda I$			Tangent of the sweep angle of the image of bound vortex; Eq.(5.1.12-5)
DMUY	$\tilde{\mu}_y$			Eq.(5.1.12-21)
DMUZ	$\tilde{\mu}_z$			Eq.(5.1.12-22)

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
INFL	f	IN/OUT	ARG	'In'-flag - 0 or 1
IOUTFL				'Out'-flag - zero if the image of bound vortex lies beyond the symmetry plane of the body - 1 otherwise
PSQR	P	IN	SUBI	Intermediate variables used in SUBI Eq.(5.1.12-18)
YCBAR	$\bar{y}_c$			(5.1.12-14)
ZCBAR	$\bar{z}_c$			(5.1.12-15)
ABAR	$\bar{a}$			(5.1.12-11)
RHO12	ρ_1^2			(5.1.12-12)
RHO22	ρ_2^2			(5.1.12-13)
ETAI1	$\eta I1$			(5.1.12-7)
ETAI2	$\eta I2$			(5.1.12-8)
ZETI1	$\zeta I1$			(5.1.12-9)
ZETI2	$\zeta I2$			(5.1.12-10)
RHO2	ρ^2			(5.1.12-23)
ELLIPS				See Eq.(5.1.12-24 or -26)
TEST				Test value;(Eq. 5.1.12-25 or -27)

Calling Subroutines SURP and FWMW

Equations

Option 1. - INFL = 0

$$\eta I = (\eta I1 + \eta I2)/2 \qquad (5.1.12\text{-}1)$$

$$\zeta I = (\zeta I1 + \zeta I2)/2 \qquad (5.1.12\text{-}2)$$

$$\cos\gamma I = -\frac{\eta I2 - \eta I1}{2eI} \qquad (5.1.12\text{-}3)$$

$$\sin\lambda I = -\frac{\zeta I2 - \zeta I1}{2eI} \qquad (5.1.12\text{-}4)$$

$$\tan\lambda I = \frac{\xi I2 - \xi I1}{2eI} \qquad (5.1.12\text{-}5)$$

$$eI = \sqrt{(\eta I2 - \eta I1)^2 + (\zeta I2 - \zeta I1)^2} \;/2.0 \qquad (5.1.12\text{-}6)$$

where

$$\eta I1 = \bar{y}_c + \frac{\bar{a}^2}{\rho_1^{\,2}}(\eta 1-\bar{y}_c) \qquad (5.1.12\text{-}7)$$

$$\eta I2 = \bar{y}_c + \frac{\bar{a}^2}{\rho_2^{\,2}}(\eta 2-\bar{y}_c) \qquad (5.1.12\text{-}8)$$

$$\zeta I1 = \bar{z}_c + \frac{\bar{a}^2}{\rho_1^{\,2}}(\zeta 1-\bar{z}_c) \qquad (5.1.12\text{-}9)$$

$$\zeta I2 = \bar{z}_c + \frac{\bar{a}^2}{\rho_2^{\,2}}(\zeta 2-\bar{z}_c) \qquad (5.1.12\text{-}10)$$

$$\bar{a} = \frac{a}{AR}(\sin^2\theta + AR^2\cos^2\theta)^{3/2} \qquad (5.1.12\text{-}11)$$

$$\rho_1^{\,2} = (\eta 1-\bar{y}_c)^2 + (\zeta 1-\bar{z}_c)^2 \qquad (5.1.12\text{-}12)$$

$$\rho_2^{\,2} = (\eta 2-\bar{y}_c)^2 + (\zeta 2-\bar{z}_c)^2 \qquad (5.1.12\text{-}13)$$

$$\bar{y}_c = a(1-AR^2)\cos^3\theta + YB \qquad (5.1.12\text{-}14)$$

$$\bar{z}_c = a\,\frac{AR^2-1}{AR}\sin^3\theta + ZB \qquad (5.1.12\text{-}15)$$

$$\cos\theta = \frac{\eta-YB}{P}AR$$

$$\sin\theta = \frac{\zeta-ZB}{P}$$

$$P = \sqrt{(\eta - YB)^2 AR^2 + (\zeta - ZB)^2} \qquad (5.1.12\text{-}18)$$

and where

$\eta 1 = \eta - e\cos\gamma$

$\eta 2 = \eta + e\cos\gamma$

$\zeta 1 = \zeta - e\sin\gamma$

$\zeta 2 = \zeta + e\sin\gamma$

<u>Option 2. - INFL = 1</u>

$$\eta I = \bar{y}_c + \frac{\bar{a}^2}{\rho^2}(\eta - \bar{y}_c) \qquad (5.1.12\text{-}19)$$

$$\zeta I = \bar{z}_c + \frac{\bar{a}^2}{\rho^2}(\zeta - \bar{z}_c) \qquad (5.1.12\text{-}20)$$

$$\tilde{\mu}_y = -\frac{\bar{a}^2}{\rho^4}\{-\sin\gamma[(\eta - \bar{y}_c)^2 - (\zeta - \bar{z}_c)^2] + 2\cos\gamma(\eta - \bar{y}_c)(\zeta - \bar{z}_c)\} \qquad (5.1.12\text{-}21)$$

$$\tilde{\mu}_z = -\frac{\bar{a}^2}{\rho^4}\{-2\sin\gamma(\eta - \bar{y}_c)(\zeta - \bar{z}_c) - \cos\gamma[(\eta - \bar{y}_c)^2 - (\zeta - \bar{z}_c)^2]\} \qquad (5.1.12\text{-}22)$$

where

$$\rho^2 = (\eta - \bar{y}_c)^2 + (\zeta - \bar{z}_c)^2 \qquad (5.1.12\text{-}23)$$

and where

$\bar{a}$, $\bar{y}_c$ and $\bar{z}_c$ are defined by Eq's 5.1.12-11, -14 and -15 respectively.

<u>Both Options 1 and 2.</u>

IF AR>1

$$ELLIPS = 2aAR \qquad (5.1.12\text{-}24)$$

$$TEST = TRMA + TRMB \qquad (5.1.12\text{-}25)$$

where

$$TRMA = \sqrt{[a\sqrt{AR^2 - 1} - (zI - z_c)]^2 + (yI - y_c)^2}$$

and

$$\text{TRMB} = \sqrt{[a\sqrt{AR^2-1} + (zI-z_c)]^2 + (yI-y_c)^2}$$

If AR<1

$$\text{ELLIPS} = 2a \tag{5.1.12-26}$$

$$\text{TEST} = \text{TRMC} + \text{TRMD} \tag{5.1.12-27}$$

where

$$\text{TRMC} = \sqrt{[a\sqrt{1-AR^2} - (yI-y_c)]^2 + (zI-z_c)^2}$$

and

$$\text{TRMD} = \sqrt{[a\sqrt{1-AR^2} + (yI-y_c)]^2 + (zI-z_c)^2}$$

and where yI and zI are the image point coordinates: ηI, ζI for the 'center' sending point, $\eta I1$, $\zeta I1$ for the 'inboard', and $\eta I2$, $\zeta I2$ for the 'outboard' sending point.

5.1.13 SUBROUTINE TIME (NN)

Functional Description

Subroutine TIME is a utility subroutine which is used to printing out the CPU- and I/O-time spent on the computer either since the beginning of the run (setting: NN = 0), or since the last call to TIME (NN = 1). In program N5KA subroutine TIME is called from MAIN for each major subroutine, so that a breakdown on the computing time per major computations is obtained for the case.

5.1.14 SUBROUTINE TKER (XO, YO, ZO, KR, BR, SGR, CGR, SGS, CGS, T1, T2, M)

Functional Description

Subroutine TKER computes the incremental oscillatory kernel, K, for one receiving-sending box combination at a time. The result is obtained in six components and is returned to the calling program, subroutine INCR , via the labeled common block DLM (see Sec. 3.2). Subroutine TKER also computes the total planar and nonplanar kernels $\tilde{K}_1$ and $\tilde{K}_2$ whenever the flag

IND is set to 0; this option is exercised by the calling subroutine FLLD. The flag IND, as well as the total kernels, are transmitted via the labeled common block KDS.

Input Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
XO	$\bar{x}$	IN	ARG	AX for center AX1 for inboard AX2 for outboard
YO	$\bar{y}$			AY for center AY1 for inboard AY2 for outboard
ZO	$\bar{z}$			AZ for center AZ1 for inboard AZ2 for outboard point of sending box; see Subroutine INCRO
KR	k_r			See Subroutine INCRO
BR	b_r, $\bar{c}/2$			
SGR				
CGR				
SGS				
CGS				
T1	T_1	OUT	TKER	$T_1 = \cos(\gamma_r - \gamma_s)$
T2	T_2			$T_2 = [(z_0\cos\gamma_r - y_0\sin\gamma_r)$ $x(z_0\cos\gamma_s - y_0\sin\gamma_s)]/(b_r/10)^2$
M	M	IN	ARG	Mach Number

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
R1	r			$r = \bar{y}^2 + \bar{z}^2$
T2P	$T_2{}^*$			See Eq. (A.8)†
BETA2	β^2			
BIGR	R			See Eq. (A.14)
K1	k_1			Eq. (A.12)
MU1	μ_1			Eq. (A.11)
IOUR	$Re(I_o)$	IN	TKER	Eq. (A.30)
IOUI	$Im(I_o)$			
JOUR	$Re(J_o)$			Eq. (A.31)
JOUI	$Im(J_o)$			
I1UR	$Re(I_1)$			Eq. (A.25)
I1UI	$Im(I_1)$			
I2UR3	$3Re(I_2)$			Eq. (A.27)
I2UI3	$3Im(I_2)$			
K1OT1	$K_1^{(s)}T_1$	OUT	TKER	Planar part } of the steady kernel
K2OT2P	$K_2^{(s)}T_2{}^*$			Nonplanar part
K1RT1 K1IT1				Planar part of } the unsteady kernel; see Sec. 3.2
K2RT2P				Nonplanar part of
K2TI2P				

† Appendix A in Part I Vol. I of this report (Reference 2)

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
KD1R	$Re(K_1)$			See Sec. 3.2
KD1J	$Im(K_1)$	OUT	TKER	
KD2R	$Re(K_2)$			
KD2I	$Im(K_2)$			

Calling Subroutines   INCRO and FLLD.

Common Blocks

Common DLM and Common KDS

Equations   See Appendix A in Part I Vol. I of this report (Reference 2)

5.1.15 SUBROUTINE TVOR (SL1, CL1, TL1, SL2, CL2, TL2, SGS, CGS, SGR, X01, X02, Y0, Z0, E, BETA, CBAR, FMACH, KR, BRE, BIM)

Functional Description

This routine calculates the normalwash at a point (x, y, z) of surface dihedral γ_r, due to a trapezoidal unsteady vortex ring of unit strength.

Input Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
SL1	$\sin\lambda_1$			λ_1 is the sweep angle of leading edge of box
CL1	$\cos\lambda_1$			
TL1	$\tan\lambda_1$			
SL2	$\sin\lambda_2$	IN	ARG	
CL2	$\cos\lambda_2$			λ_2 is the sweep angle of trailing edge of box
TL2	$\tan\lambda_2$			
SGS	$\sin\gamma_s$			Sine of dihedral angle at sending point

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
CGS	$\cos\gamma_s$			Cosine of dihedral angle at sending point
SGR	$\sin\gamma_r$			Sine of dihedral angle of receiving point
CGR	$\cos\gamma_r$			Cosine of dihedral angle of receiving point
X01	$x-\xi_1$			Distance in x-direction from receiving point to leading edge of box
X02	$x+\xi_2$			ξ_2 indicates trailing edge of box
YO	$y-\eta$	IN	ARG	Distance in y-direction from receiving point to center of box
ZO	$y-\xi$			Distance in z-direction from receiving point to center of box
E	e			
BETA	β			$\sqrt{1 - M^2}$
CBAR	$\bar{c}$			Length of reference chord
FMACH	M			Mach Number
KR	k_r			Reduced frequency
BRE	Re(B)	OUT	ARG	Real part of the normalwash (See Eq. 5.1.15-1)
BIM	Im(B)	OUT	ARG	Imaginary part of the normalwash (See Eq. 5.1.15-1)

Calling Subroutines DZY

Called Subroutines and common blocks SNPDF, FLLD

Equations

$$B = \frac{BS}{2e\Delta x} - \frac{\Delta B}{48\pi} \qquad (5.1.15\text{-}1)$$

where

$$BS = D^{(s)}[(\lambda_1, x_{01}), (\gamma_r, \gamma_s, Z_o, Y_o, e, \beta, \Delta x)]$$

$$- D^{(s)}[(\lambda_2, x_{02}), (\gamma_r, \gamma_s, Z_o, Y_o, e, \beta, \Delta x)]$$

$D^{(s)}$ - is calculated in subroutine SNPDF

$$\Delta x = \xi_2 - \xi_1$$

$\underline{r_i \text{ and } r_o \geq \frac{e}{4}}$

$$\Delta B' = \Delta\tilde{K}_{d_i} + 4\Delta\tilde{K}_{d_c} + \Delta\tilde{K}_{d_o}$$

$\underline{r_i \text{ or } r_o < \frac{e}{4}}$

$$\Delta B' = 6\Delta\tilde{K}_{d_c}$$

$\underline{r_c \leq \frac{e}{4}}$

$$\Delta B' = 3\Delta\tilde{K}_{d_i} + 3\Delta\tilde{K}_{d_o}$$

where

$$\Delta\tilde{K}_d = \frac{\Delta K_{d1}}{r^2} + \frac{\Delta K_{d2}}{r^4}$$

$$r_i^2 = (y_o + e\cos\gamma_s)^2 + (z_o + e\cos\gamma_s)^2$$

$$r_c^2 = y_o^2 + z_o^2$$

$$r_o^2 = (y_o - e\cos\gamma_s)^2 + (z_o - e\cos\gamma_s)^2$$

and

ΔK_{d1} and ΔK_{d2} are calculated by FLLD

5.1.16 SUBROUTINE WRTFMF (IUNT, NBE, FZ, FY, EMZ, EMY)

Functional Description

This routine writes the NBE elements of the complex arrays FZ, FY, EMZ, and EMY with a format on I/O unit IUNT.

Input Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
IUNT				I/O unit on which the data is to be written
NBE				Number of complex words in each array
FZ		IN	ARG	Array to be written
FY				Array to be written
EMZ				Array to be written
EMY				Array to be written

Calling Subroutine BFM

5.1.17 SUBROUTINE WRTFMU (IUNT, MODE, NBE, FZ, FY, EMZ, EMY)

Functional Description

This routine writes MODE and the NBE elements of the complex arrays FZ, FY, EMZ, EMY without a format on the I/O unit IUNT.

Input Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
IUNT				I/O unit on which the data is to be written
MODE				The mode number to be written
FZ		IN	ARG	Array to be written
FY				Array to be written
EMZ				Array to be written
EMY				Array to be written

Calling Subroutine BFM

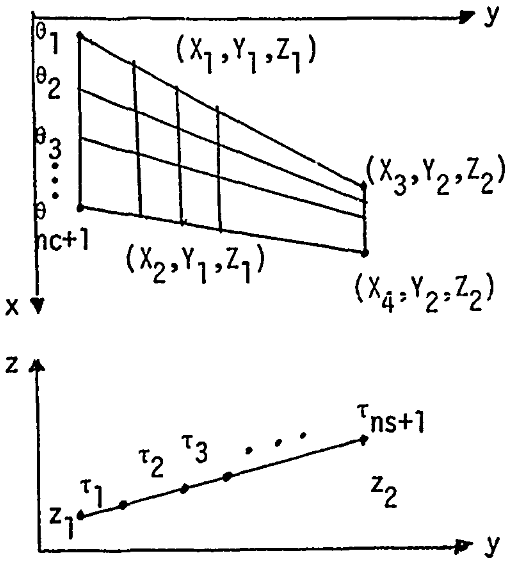
5.2 Segment 2

5.2.1 SUBROUTINE DATA

Functional Description

Subroutine DATA reads the x-, y- and z-coordinates that define each panel, interference body and slender body for the case considered. It also readsthe fractional chordwise and spanwise divisions for each panel and calculates arrays of geometry that define the sending- and receiving control points on the lifting surfaces. In the case when bodies are also present, similar geometry arrays are calculated for all interference body elements, and all slender body elements. These basic data arrays are saved in the Blank Common Block for use in the subsequent calculations.

Input Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
X1	X_1			Panel coordinates 
X2	X_2			
X3	X_3			
X4	X_4			
Y1	Y_1			
Y2	Y_2	IN	card	
Z1	Z_1			
Z2	Z_2			
NC	nc			Number of chordwise boxes
NS	ns			Number of spanwise strips
NAB(10)				Associated bodies
YIN(100)	$Y_1^{(p)}$	OUT	DATA	y-coordinate of inboard edge of panel p

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
ZIN(100)	$Z_1^{(p)}$			z-coordinate of inboard edge of panel
NASB(200)				Array of all associated bodies for all panels
NAS(100)				Array of the number of associated bodies for all panels
NCARAY(100)	nc	OUT	DATA	Array of the number of chordwise boxes for all panels
NSARAY(100)	ns			Array of the number of spanwise strips for all panels
NBARAY(100)	nba			$nba_p = \sum_{i=1}^{p} nc_i \; ns_i$ where p is the panel number
TH(50)	$e_c^{(p)}$	IN	card	Fractional chordwise divisions for panel p
TAU(50)	$\tau_s^{(p)}$			Fractional spanwise divisions for panel p
GMA(100)	$\gamma^{(p)}$			Dihedral angle of panel p
X(500)	$x^P_{c,s}$ and x^I_k			3/4-chord x-coordinate of all boxes and x-coordinate of interference body section midpoints - see Eqs.(5.2.1-1 and-2)
XI1(500)	$\xi P1_{c,s}$			1/4-chord coordinates of inboard- and outboard edge of panel boxes - see Eqs.(5.2.1-3 and -4)
XI2(500)	$\xi P2_{c,s}$	OUT	DATA	
DELX(500)	$\Delta x P_{c,s}$ and $\Delta x I_k$			Average chordlength of boxes and Δx of interference body sections - see Eqs.(5.2.1-7 and -8)

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
YS(210)	yP_s			y-coordinate of centerline of panel strips - Eq.(5.2.1-9)
ZS(210)	zP_s			z-coordinates of centerline of panel strips - Eq.(5.2.1-10)
DYS(200)	Δy_s			See Eqs. (5.2.1-12) and (5.2.1-13)
DZS(200)	Δz_s			
EE(200)	ϵ_s			Half-width of panel strips - Eq.(5.2.1-11)
CS(200)	c_s	OUT	DATA	Average chord of panel strips - Eq.(5.2.1-16)
SG(200)				sine and cosine of dihedral angle for strips - Eqs. (5.2.1-14 and -15)
CG(200)				
XIJ(200)				x-coordinate of leading edge of strip centerlines
GMAR(200)	γ_s			Dihedral angle of strips in radians
XLAM(500)	$\tan\lambda_{c,s}$			Tangent of the sweep angle of the 1/4-chordline of all lifting surface boxes - see Eq.(5.2.1-13a)
ZC	z_c			z-coordinate of body centerline
YX	y_c			y-coordinate of body centerline
RAD	a			Average half-width of body
AR		IN	card	Cross-sectional aspect ratio of body
NBE				No. of interference body elements
NSBE				No. of slender body elements
NZY				z-y orientation flag

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
NRI				Interference body element radius flag (RI-flag)
NRS				Slender body element radius flag (RS-flag)
NSH				Number of $\Delta\eta$ - $\Delta\zeta$ pairs for body
NT1	$N\theta_1$	IN	card	Number of θ_1's for body
NT2	$N\theta_2$			Number of θ_2's for body
ZB(10)				Array of z-coordinates of body centerlines
YB(10)				Array of y-coordinates of body centerlines
AVR(10)				Array of average characteristic half widths,
ARB(10)		OUT	DATA	Array of aspect ratios,
NBEA(10,2)				1. No. of interference body elements 2. z-y flags,
NSBEA(10)				No. of slender body elements - for all bodies
XII(100)	ξI			x-coordinates of interference body element endpoints
RI(100)	RI			Radii of interference body element endpoints
XIS(100)	ξS	IN	card	x-coordinates of slender body element endpoints
RS(100)	RS			Radii of slender body element endpoints

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
TH1(24)	$\theta 1_{\mu}$	IN	card	Angular orientation of point μ on body, first set
TH2(24)	$\theta 2_{\mu}$			Angular orientation of point μ on body, second set
TH1A(100)		OUT	DATA	Array of $\theta 1_{\mu}$'s for all bodies
TH2A(100)				Array of $\theta 2_{\mu}$'s for all bodies
L1 L2 L3 L4 L5 L6		IN	card	First and last interference body elements associated with $\theta 1_{\mu}$
IFLA(30,2)		OUT	DATA	1. Array of 'first' interference body elements, 2. Array of 'last' interference body elements,
NFL(10)				Array of the number of pairs of 'first- and last' elements - for all bodies
NCD1 CD2 CD3		IN	card	Panel no. for $\Delta\eta$, $\Delta\zeta$ y-shift of panel z-shift of panel
NCD4 CD5 CD6				another set of the above 3 items
NOBODY(50)				Array of the body no.'s,
NPANEL(50)				Array of the panel no's for the $\Delta\eta$, $\Delta\zeta$ shift

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
DETA(50)	$\Delta\eta$			Array of the y-shifts for panels
DZET(50)	$\Delta\zeta$			Array of the z-shifts for panels
RIA(100)				Array of the radii of all interference body element midpoints
XLE(10)				Eq.(5.2.1-17) when NBE≠0. If NBE=0, but NSBE≠0, the slender body leading- and trailing x-coordinates are used
XTE(10)				Eq.(5.2.1-18)
XIS1(200)	$\xi S1_t$	OUT	DATA	Array of the x-coordinates of leading- and trailing edges (respectively) of all slender body elements; see Eq's (5.2.1-20 and -21)
XIS2(200)	$\xi S2_t$			
A0(200)	a_{o_t}			Array of the average radii of all slender body elements; see Eq.(5.2.1-22)
AOP(200)	a_{o_t}			Array of the first derivatives of a_{o_t} for all slender body elements; see Eq.(5.2.1-23)

Calling Subroutine: MAIN

Equations

3/4-chord x-coordinate for all boxes of panel:

$$xP_{c,s} = \frac{(\tau_s + \tau_{s+1})}{2} \left[\left(\frac{1}{4}\theta_c + \frac{3}{4}\theta_{c+1} \right)(\chi_4 - \chi_3 - \chi_2 + \chi_1) + (\chi_3 - \chi_1) \right]$$

$$+ \left(\frac{1}{4}\theta_c + \frac{3}{4}\theta_{c+1} \right)(\chi_2 - \chi_1) + \chi_1 \qquad (\ 5.2.1\text{-}1)$$

x-coordinate of interference-body section midpoint:

$$xI_k = \frac{\xi I_{k+1} + \xi I_k}{2} \tag{5.2.1-2}$$

1/4-chord x-coordinate of inboard edge of panel boxes:

$$\xi P1_{c,s} = \tau_s \, BR + CR \tag{5.2.1-3}$$

1/4-chord x-coordinate of outboard edge of panel boxes:

$$\xi P2_{c,s} = \tau_{s+1} \, BR + CR \tag{5.2.1-4}$$

where

$$BR = (\tfrac{3}{4}\,\theta_c + \tfrac{1}{4}\,\theta_{c+1})(X_4 - X_3 - X_2 + X_1) + (X_3 - X_1)$$

$$CR = (\tfrac{3}{4}\,\theta_c + \tfrac{1}{4}\,\theta_{c+1})(X_2 - X_1) + X_1$$

x-coordinate of leading and trailing edges of interference-body sections:

$$\xi I1_k = \xi I_k \tag{5.2.1-5}$$

$$\xi I2_k = \xi I_{k+1} \tag{5.2.1-6}$$

Average chord-length of panel boxes:

$$\Delta XP_{c,s} = \frac{\tau_{s+1} + \tau_s}{2} \; [(\theta_{c+1} - \theta_c)(X_4 - X_3 - X_2 + X_1)]$$

$$+ (\theta_{c+1} - \theta_c)(X_3 - X_1)] \tag{5.2.1-7}$$

Δx for interference body-sections:

$$\Delta XI_k = \xi I2_k - \xi I1_k \tag{5.2.1-8}$$

y-coordinate of centerline of panel strips:

$$yP_s = \frac{\tau_s + \tau_{s+1}}{2} (Y_2 - Y_1) + Y_1 \tag{5.2.1-9}$$

z-coordinate of centerline of panel strips:

$$zP_s = \frac{\tau_s + \tau_s + 1}{2} (Z_2 - Z_1) + Z_1 \tag{5.2.1-10}$$

Half-width of panel strips

$$e_s = \frac{1}{2} \sqrt{\Delta y_s^{\,2} + \Delta z_s^{\,2}} \tag{5.2.1-11}$$

where

$$\Delta y_s = (\tau_{s+1} - \tau_s)\,(Y_2 - Y_1) \tag{5.2.1-12}$$

$$\Delta z_s = (\tau_{s+1} - \tau_s)\,(Z_2 - Z_1) \tag{5.2.1-13}$$

$$\tan\lambda_{c,s} = (\xi P2_{c,s} - \xi P1_{c,s})/(2e_s) \tag{5.2.1-13a}$$

Sine and cosine of dihedral angle of panel strips

$$\sin\gamma_s = \Delta z_s/2e_s \tag{5.2.1-14}$$

$$\cos\gamma_s = \Delta y_s/2e_s \tag{5.2.1-15}$$

Average chord of panel strips

$$c_s = \frac{\tau_s + \tau_{s+1}}{2} (x_4 - x_3 - x_2 + x_1) + (x_2 - x_1) \tag{5.2.1-16}$$

Leading and trailing edge coordinates of bodies

$$X_{L.E.} = \xi I1_k \quad , \quad k = 1 \tag{5.2.1-17}$$

$$X_{T.E.} = \xi I2_k \quad , \quad k = NBE \tag{5.2.1-18}$$

x-coordinate of slender body section midpoint

$$xS_t = (\xi T_t + \xi S_{t+1})/2 \tag{5.2.1-19}$$

x-coordinates of leading and trailing edges of slender body sections

$$\xi S1_t = \xi S_t \tag{5.2.1-20}$$

$$\xi S2_t = \xi S_{t+1} \tag{5.2.1-21}$$

Average radius of slender body sections:

$$a_{o_t} = \frac{RS_t + RS_{t+1}}{2} \tag{5.2.1-22}$$

First derivative of a_o:

$$a_{o_t}' = \frac{RS_{t+1} - RS_t}{\Delta xS_t} \tag{5.2.1-23}$$

where

$$\Delta xS_t = \xi S2_t - \xi S1_t \tag{5.2.1-24}$$

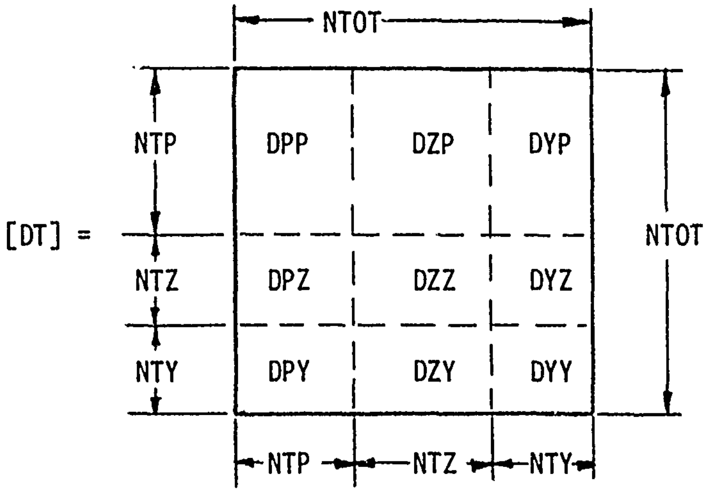
5.3 Segment 3

5.3.1 SUBROUTINE GEND (NPRINT, NTAPE, WORK)

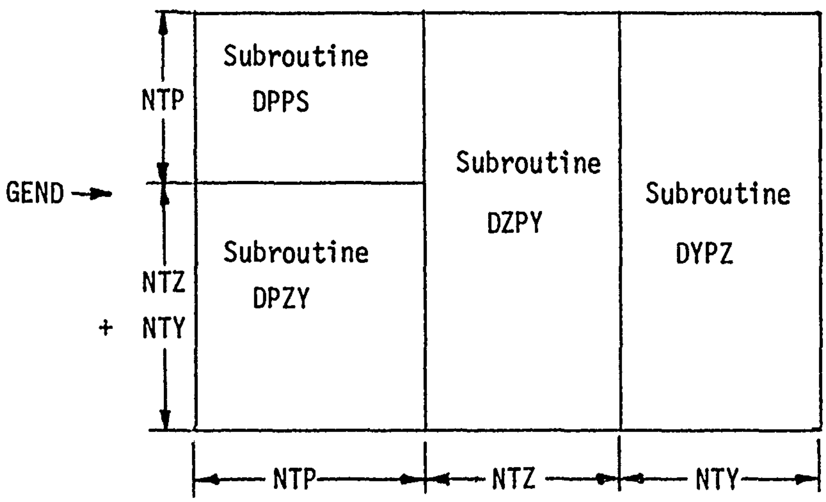
Functional Description

Subroutine GEND generates the nine submatrices of the [DT] matrix, then assembles the [DT] matrix and writes it on logical tape unit no. ITP8 in row order. The submatrices generated by GEND, and the major subroutines that compute these, are shown in the diagrams below.

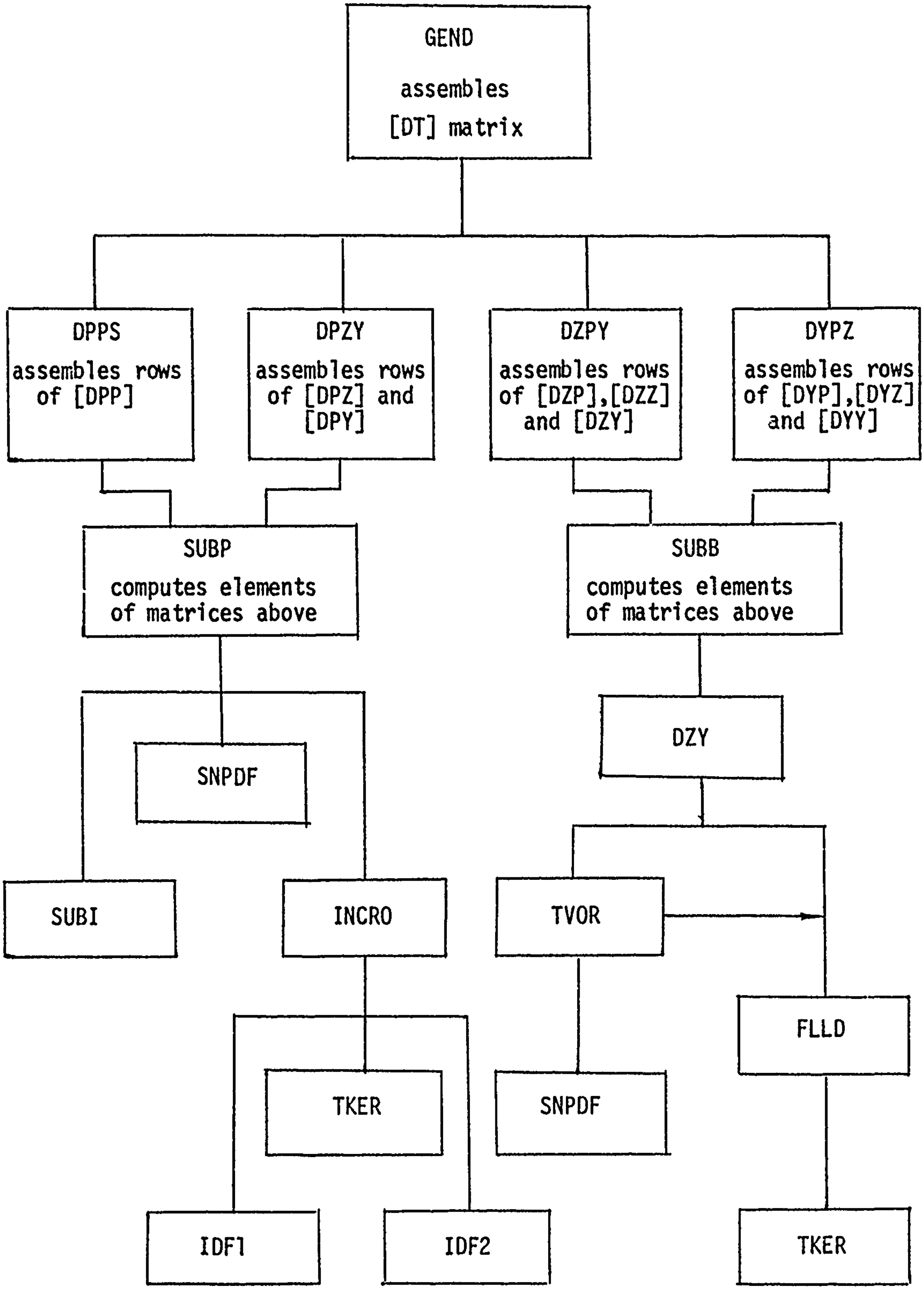
DT Matrix Structure



Major Subroutines Generating DT



General Flow Chart - Subroutine GEND



Input Output Variables

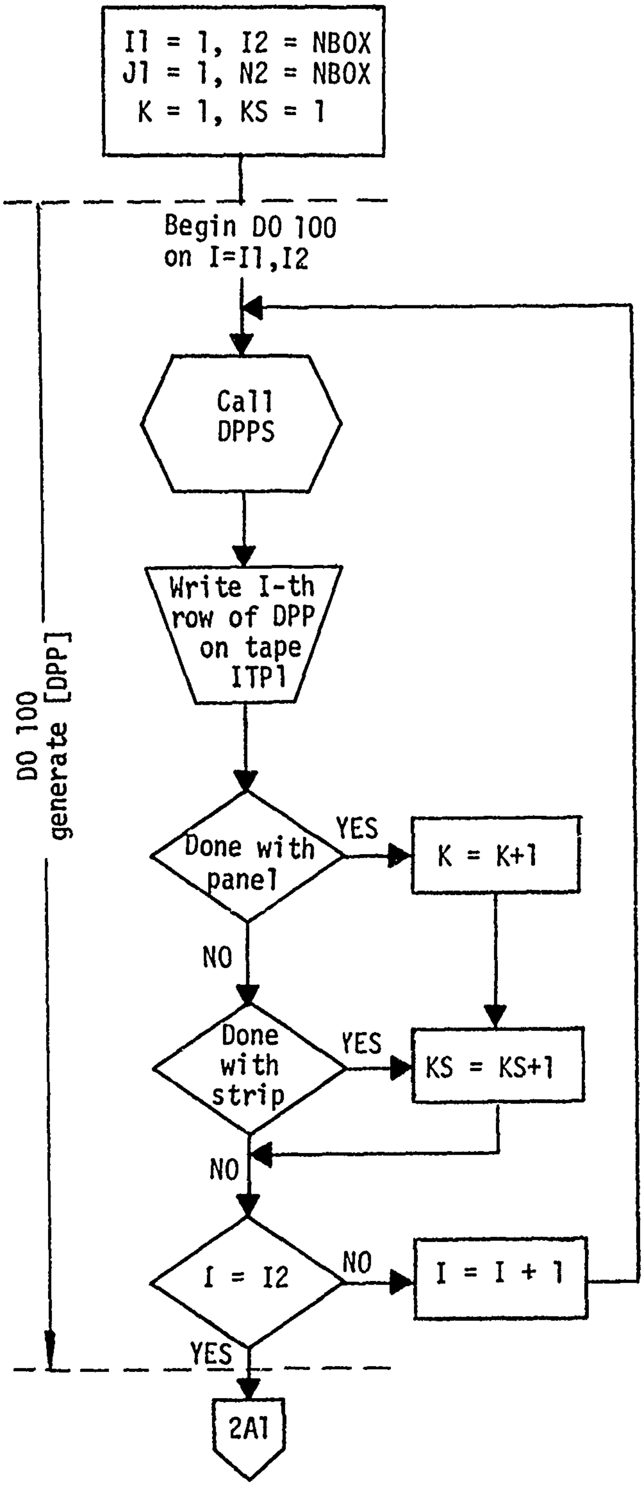
MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
NPRINT		IN	ARG	Print flag for DT matrix
NTAPE(20)		IN	ARG	Logical tape unit array
WORK		IN/OUT		Complex work array
DT(500)	DT.	OUT	GEND	One row of the complex DT matrix - also used for temporary storage of rows of all submatrices of DT, except those of DPZ and DPY.
DPZ(500)	DPZ			One row of the complex DPZ matrix
DPY(500)	DPY			One row of the complex DPY matrix
FLND	δ	IN	Blank Common Block	Floating point variable for the integer input δ
FLNE	ε			Floating point variable for the integer input ε
FL,REFC	$\bar{c}$			Reference chord
I1 I2		OUT	GEND	Do-loop delimiters for the number of rows of a particular submatrix to be computed within the do loop
J1 J2				Do-loop delimiters for the number of elements in one row of a particular submatrix to be computed within do-loop.
ICOUNT				Running index of the section number of the receiving body, KB, with θ_1 distribution.
IFL				The number of sections of body KB with θ_1 distribution.
NZYKB				z-y flag of body KB
NZYSV				z-y flat of body preceding the present body KB; 0 for KB=1
IFIRST				Sequence number of the first element in the current section of body KB with θ_1 distribution
ILAST				Sequence no. of the last element in the current section of body KB with θ_1 distribution
NYFLAG				Internal flag selecting correct do-loop for bodies

Calling Subroutine   MAIN

Called Subroutines and Common Blocks

Subroutines DPPS, DPZY, DZYP and DYPZ, and the Blank Common Block.

Semi-Detailed Block Diagram of Subroutine GEND



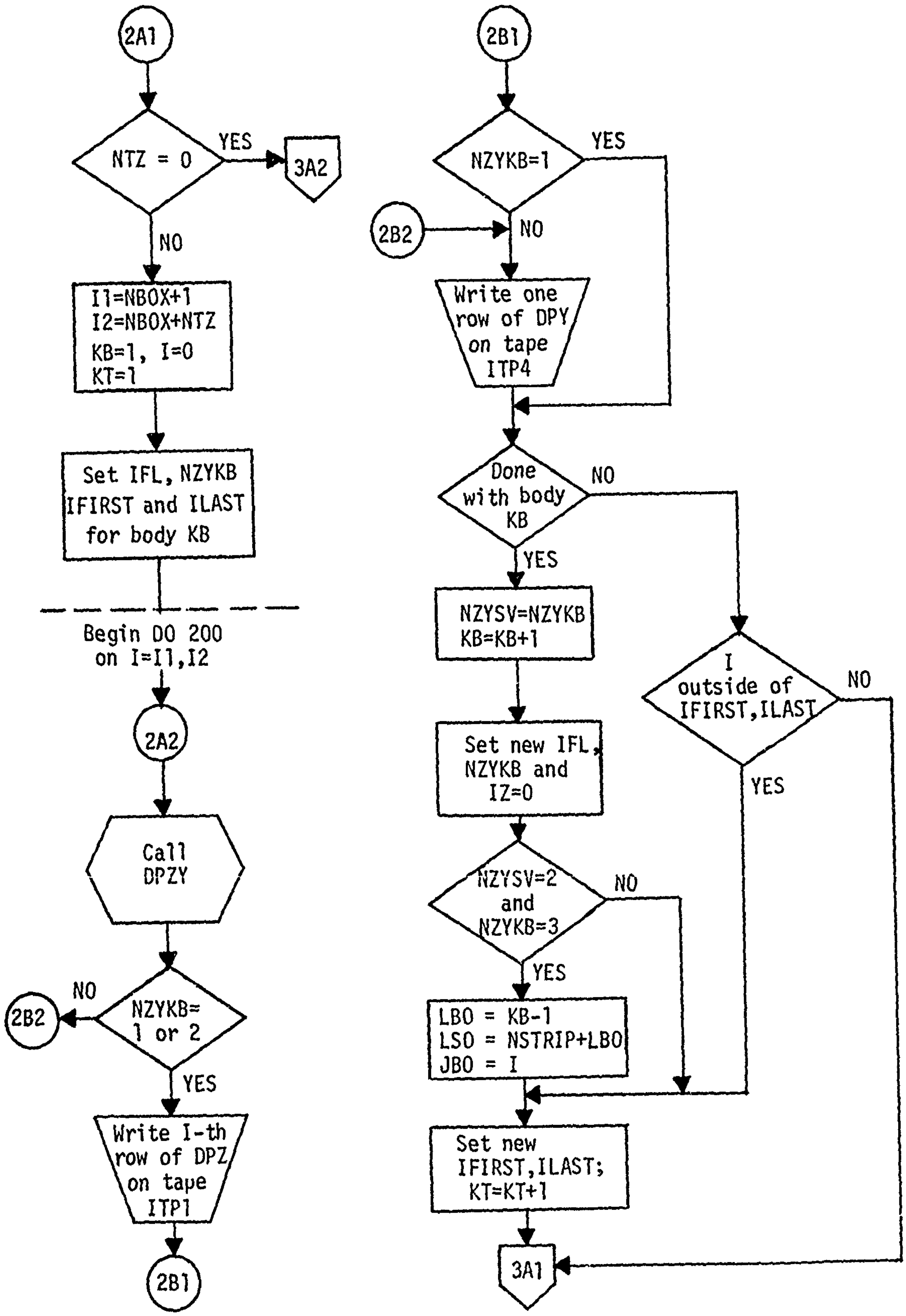
Legend

Receiving indices 'r'

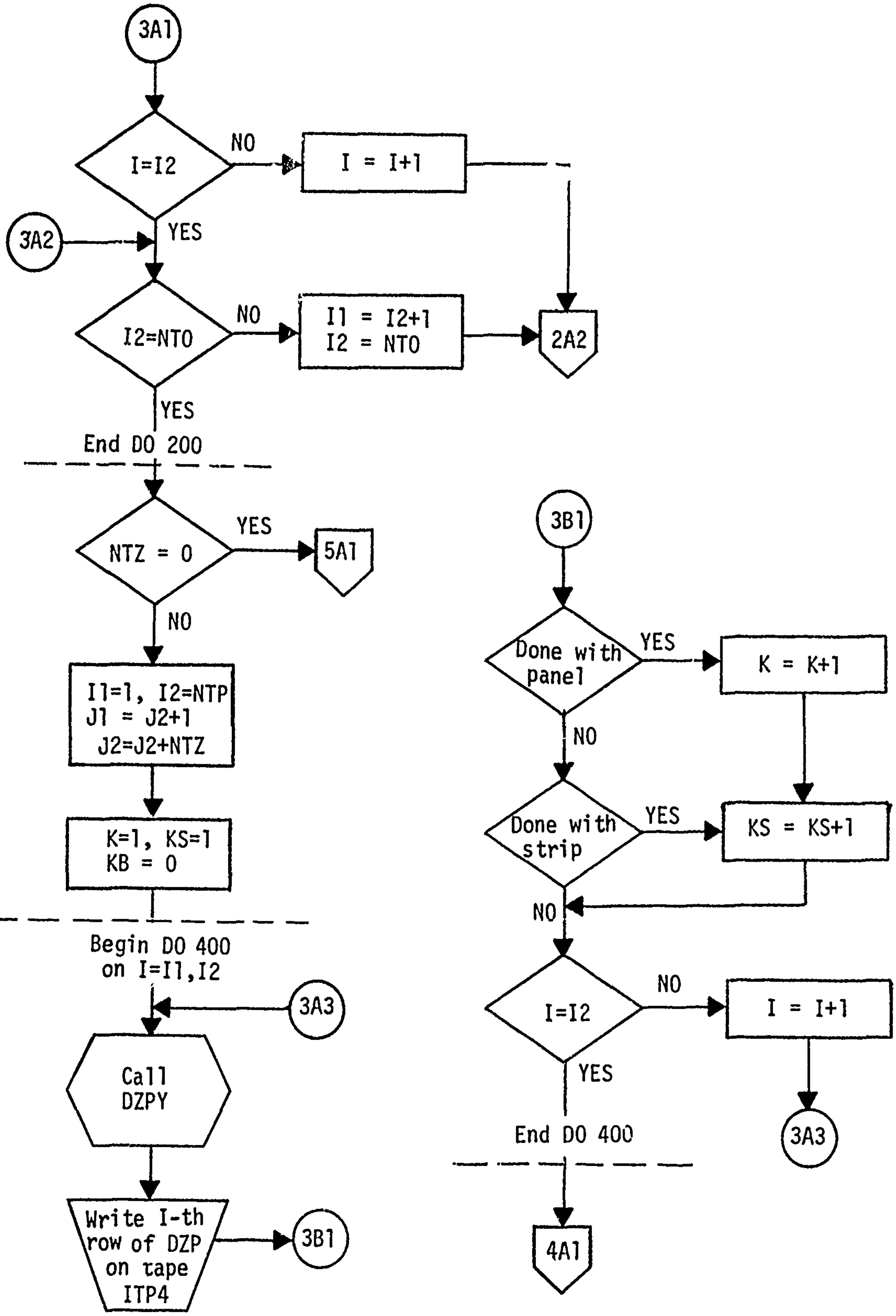
I	-	r point
K	-	r panel
KS	-	r strip (or equivalent for body)
KB	-	r body (if any)
KT	-	index of the array of the 'first-and-last elements' for θ_1
IZ	-	r body element

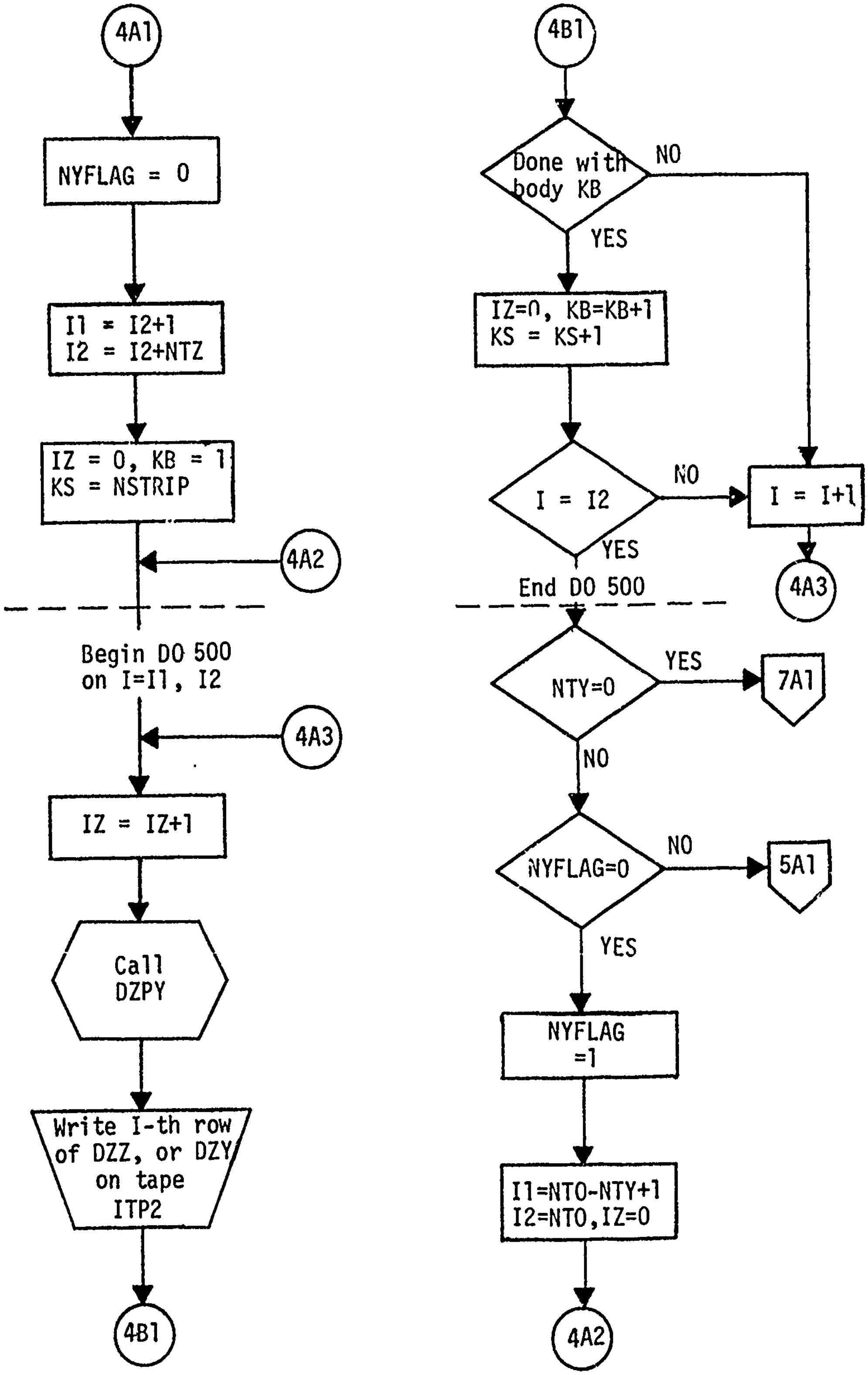
Sending indices 's'

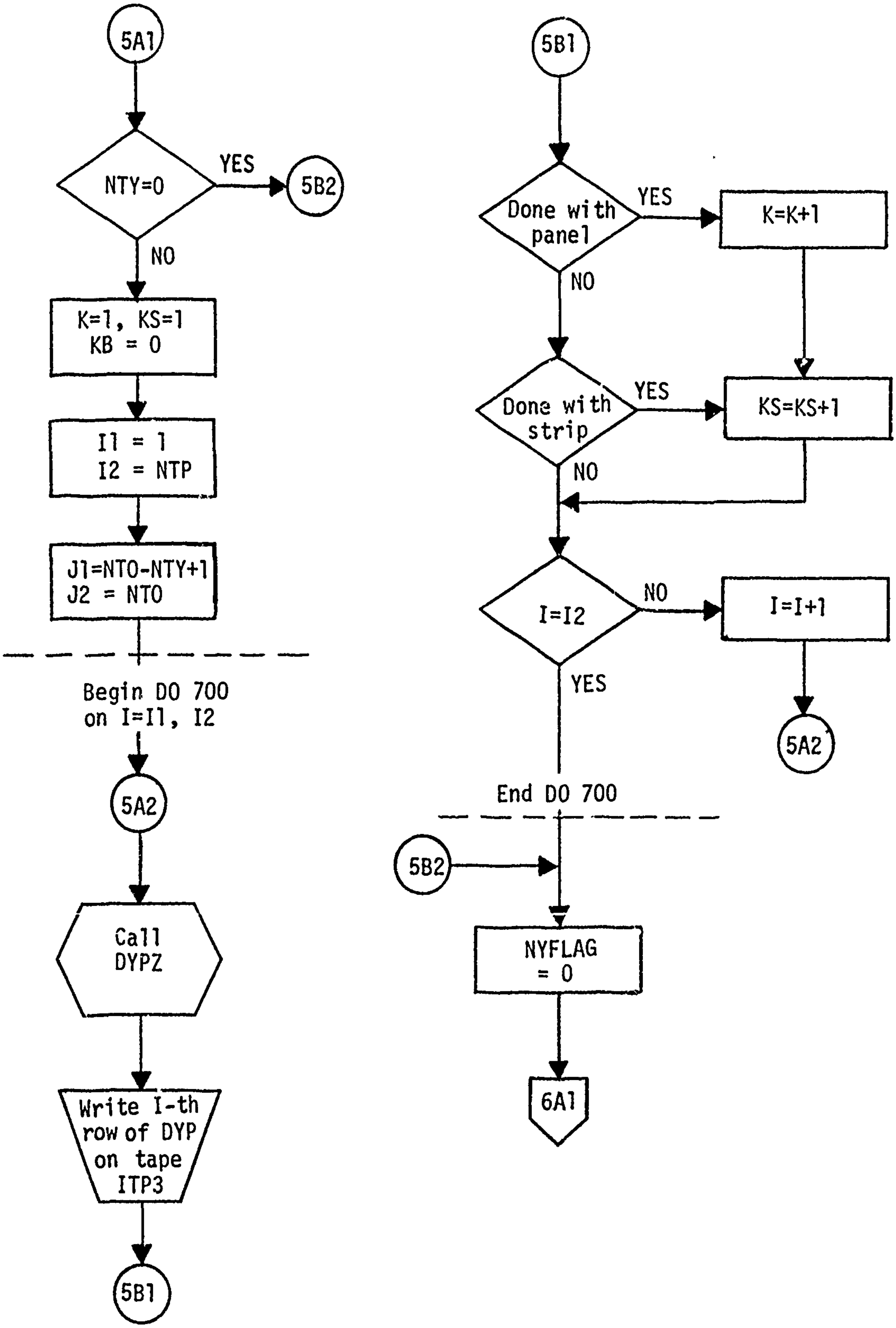
J	-	s point
L	-	s panel
LS	-	s strip (or equivalent for body)
LB	-	s body

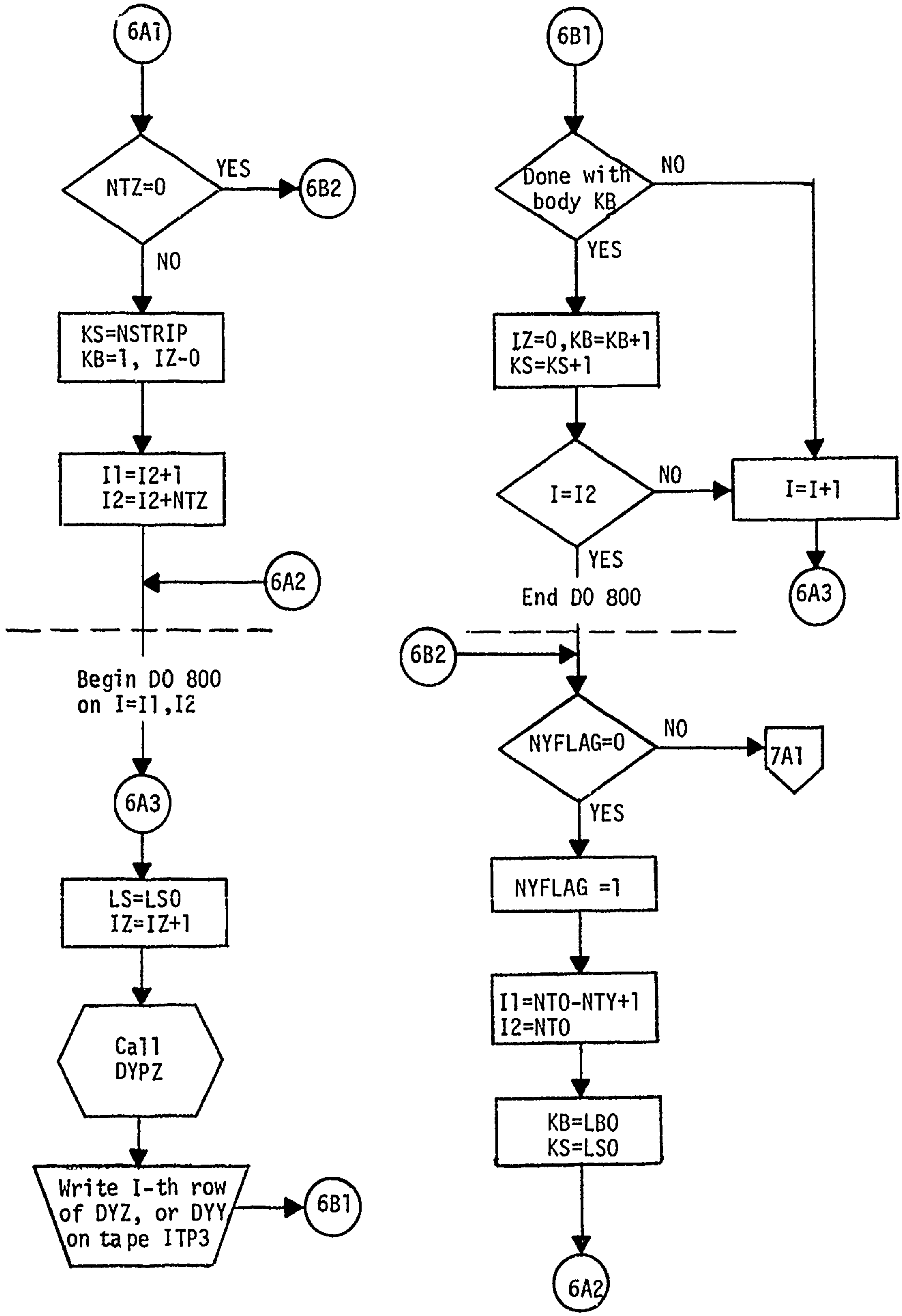


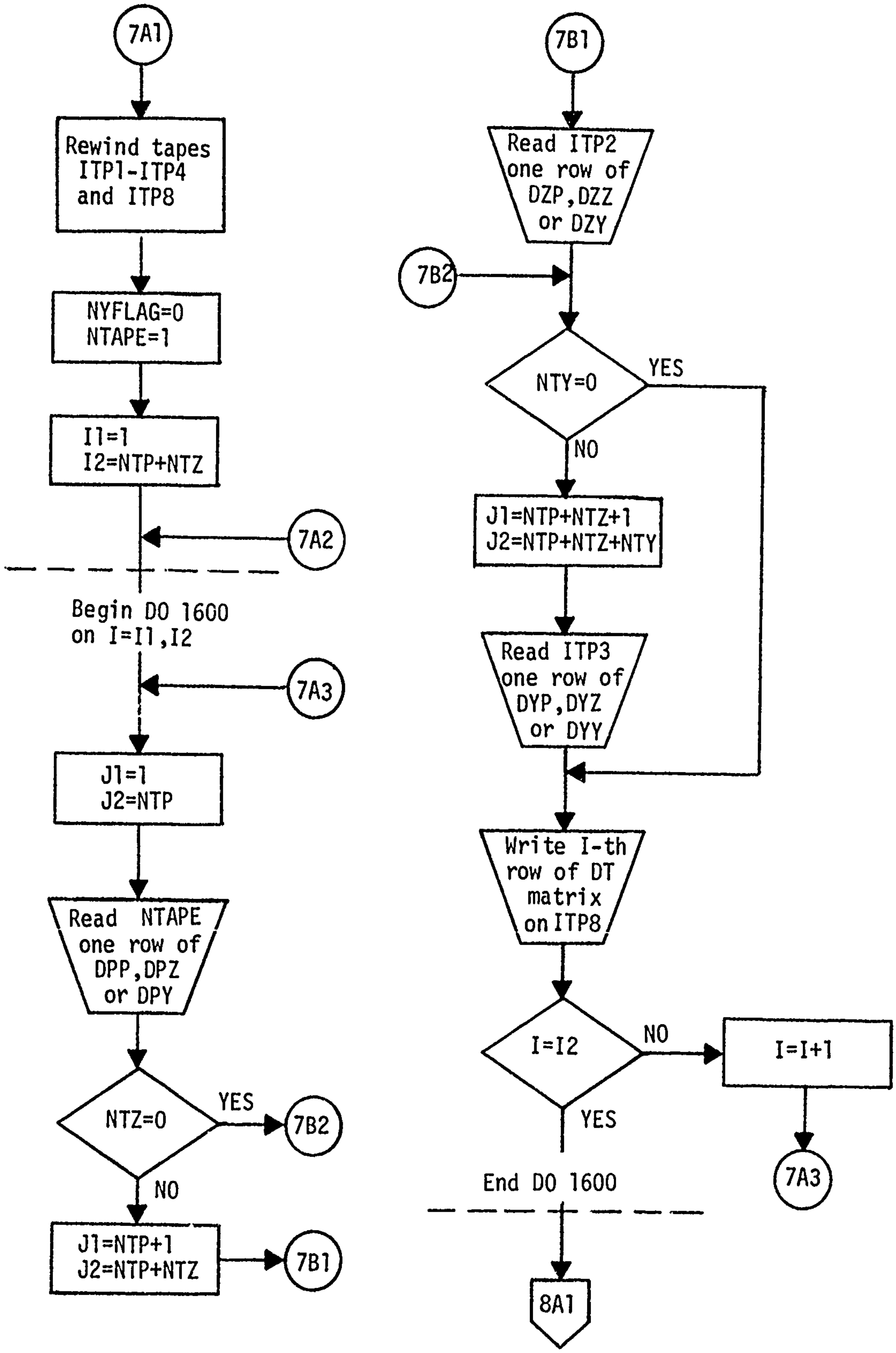
2A1
NTZ = 0
YES
3A2
NO
I1=NBOX+1
I2=NBOX+NTZ
KB=1, I=0
KT=1
Set IFL, NZYKB
IFIRST and ILAST
for body KB
Begin DO 200
on I=I1,I2
2A2
Call
DPZY
NZYKB=
1 or 2
NO
2B2
YES
Write I-th
row of DPZ
on tape
ITP1
2B1
2B1
NZYKB=1
YES
2B2
NO
Write one
row of DPY
on tape
ITP4
Done
with body
KB
NO
YES
NZYSV=NZYKB
KB=KB+1
I
outside of
IFIRST,ILAST
NO
YES
Set new IFL,
NZYKB and
IZ=0
NZYSV=2
and
NZYKB=3
NO
YES
LBO = KB-1
LSO = NSTRIP+LBO
JBO = I
Set new
IFIRST,ILAST;
KT=KT+1
3A1


3A1
I=I2
NO
I = I+1
YES
3A2
I2=NTO
NO
I1 = I2+1
I2 = NTO
2A2
YES
End DO 200
NTZ = 0
YES
5A1
NO
I1=1, I2=NTP
J1 = J2+1
J2=J2+NTZ
K=1, KS=1
KB = 0
Begin DO 400
on I=I1,I2
3A3
Call
DZPY
Write I-th
row of DZP
on tape
ITP4
3B1
3B1
Done with
panel
YES
K = K+1
NO
Done with
strip
YES
KS = KS+1
NO
I=I2
NO
I = I+1
YES
End DO 400
3A3
4A1

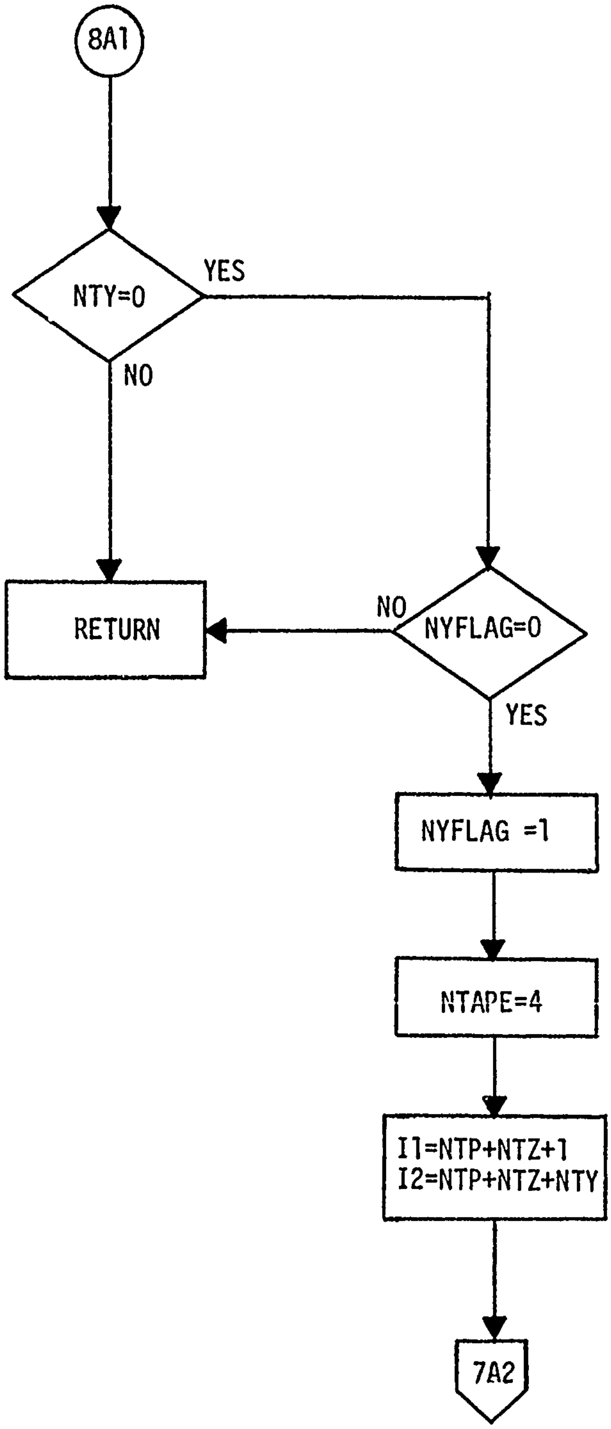

4A1
NYFLAG = 0
I1 = I2+1
I2 = I2+NTZ
IZ = 0, KB = 1
KS = NSTRIP
4A2
Begin DO 500
on I=I1, I2
4A3
IZ = IZ+1
Call
DZPY
Write I-th row
of DZZ, or DZY
on tape
ITP2
4B1
4B1
Done with
body KB
NO
YES
IZ=0, KB=KB+1
KS = KS+1
I = I2
NO
I = I+1
YES
4A3
End DO 500
NTY=0
YES
7A1
NO
NYFLAG=0
NO
5A1
YES
NYFLAG
=1
I1=NTO-NTY+1
I2=NTO,IZ=0
4A2


5A1
NTY=0
YES
5B2
NO
K=1, KS=1
KB = 0
I1 = 1
I2 = NTP
J1=NTO-NTY+1
J2 = NTO
Begin DO 700
on I=I1, I2
5A2
Call
DYPZ
Write I-th
row of DYP
on tape
ITP3
5B1
5B1
Done with
panel
YES
K=K+1
NO
Done with
strip
YES
KS=KS+1
NO
I=I2
NO
I=I+1
YES
5A2
End DO 700
5B2
NYFLAG
= 0
6A1


6A1
NTZ=0
YES
6B2
NO
KS=NSTRIP
KB=1, IZ-0
I1=I2+1
I2=I2+NTZ
6A2
Begin DO 800
on I=I1,I2
6A3
LS=LSO
IZ=IZ+1
Call
DYPZ
Write I-th row
of DYZ, or DYY
on tape ITP3
6B1
6B1
Done with
body KB
NO
YES
IZ=0,KB=KB+1
KS=KS+1
I=I2
NO
I=I+1
YES
End DO 800
6A3
6B2
NYFLAG=0
NO
7A1
YES
NYFLAG =1
I1=NTO-NTY+1
I2=NTO
KB=LBO
KS=LSO
6A2



7A1
Rewind tapes ITP1-ITP4 and ITP8
NYFLAG=0 NTAPE=1
I1=1 I2=NTP+NTZ
7A2
Begin DO 1600 on I=I1,I2
7A3
J1=1 J2=NTP
Read NTAPE one row of DPP,DPZ or DPY
NTZ=0
YES
7B2
NO
J1=NTP+1 J2=NTP+NTZ
7B1
7B1
Read ITP2 one row of DZP,DZZ or DZY
7B2
NTY=0
YES
NO
J1=NTP+NTZ+1 J2=NTP+NTZ+NTY
Read ITP3 one row of DYP,DYZ or DYY
Write I-th row of DT matrix on ITP8
I=I2
NO
I=I+1
YES
7A3
End DO 1600
8A1


8A1
NTY=0
YES
NO
RETURN
NO
NYFLAG=0
YES
NYFLAG =1
NTAPE=4
I1=NTP+NTZ+1
I2=NTP+NTZ+NTY
7A2

5.3.2 SUBROUTINE DPPS (K, KS, I, J1, J2, SGR, CGR, REFC, FMACH, YS, ZS, NBARAY, NCARAY, DT, WORK)

Functional Description

Subroutine DPPS prepares the variables necessary for the computation of one row of the DPP-submatrix and calls subroutine SUBP in a do-loop for each element of this row. The resulting matrix row, DT(500), is returned to subroutine GEND via the argument list of subroutine DPPS.

Input-Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
K		IN	ARG	Panel number in which the receiving point 'i' lies
KS		IN	ARG	Strip number in which the receiving point 'i' lies
I	i	IN/OUT	ARG	Receiving point index
J1		IN	ARG	Do-loop delimiters for the number of elements in one row of the DPP matrix
J2		IN	ARG	
SGR	$\sin\gamma_r$	IN/OUT	ARG	sine, and
CGR	$\cos\gamma_r$	IN/OUT	ARG	cosine of the dihedral angle of receiving strip
REFC	$\bar{c}$	IN	ARG	Reference chord
FMACH	M	IN	ARG	Mach Number
YS(210)	y	IN	ARG	y-array See Sec. 5.2.1
ZS(210)	z	IN	ARG	z-array
NBARAY(100)		IN	ARG	See Blank Common, Sec. 3.1
NCARAY(100)		IN	ARG	
DT(500)		IN/OUT	ARG	One row of the DPP matrix
WORK		IN/OUT	ARG	Complex work array

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
IO NBXS NCPNB		OUT	DPPS	See INCRO; Sec. 5.1.7
IR				Index of sending point
YREC	y			y-coordinate } of receiving point i
ZREC	z			z-coordinate } of receiving point i
L				Panel number } in which the sending point 'j' lies
LS				Strip number } in which the sending point 'j' lies

<u>Calling Subroutine</u> GEND

<u>Called Subroutine</u> SUBP

5.3.3 <u>SUBROUTINE DPZY (KB, KT, IZ, I, J1, J2, IFIRST, ILAST, REFC, FMACH, YB, ZB, AVR, ARB, TH1A, TH2A, NT12, NBARAY, NCARAY, NZYKB, DPZ, DPY, WORK)</u>

<u>Functional Description</u>

Subroutine DPZY prepares the variables necessary for the computation of one row of either the DPZ, or the DPY submatrix and calls subroutine SUBP in a double do-loop for each element of the row, to perform the summation given in Eqs (5.3.3-1 and -9). The resulting matrix row is returned to the calling subroutine GEND via the argument list of DPZY.

<u>Input-Output Variables</u>

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
KB		IN/OUT	ARG	Body number in which the receiving point 'i' lies
IZ				Body element number of body KB in which 'i' lies
I	i			Receiving point index

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
J1				Do-loop delimiters for the number of elements in one row of either the DPZ, or DPY matrix
J2				
IFIRST				See Subroutine GEND
ILAST				Sec. 5.3.1
REFC	$\bar{c}$			Reference chord
FMACH	M			
YB(10)	$y_c^{(b)}$			See Blank Common Sec. 3.1
ZB(10)	$z_c^{(b)}$	IN	ARG	
AVR(10)	$a^{(b)}$			
ARB(10)	$AR^{(b)}$			
TH1A(100)	$\theta1_\mu$			
TH2A(100)	$\theta2_\mu$			
NT12(10,2)				
NBARAY(100)				See Blank Common Sec. 3.1
NCARAY(100)				
NZYKB				z-y flag of receiving body
DPZ(500)		IN/OUT		One row of the DPZ matrix (5.3.3-1)
DPY(500)				One row of the DPY matrix Eq. (5.3.3-9)
IX1		OUT	DPZY	Do-loop delimiters in the summation of Eq.(5.3.3-1)
IX2				
WORK		IN/OUT	ARG	Complex work array

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
DELTH	$\Delta\theta$	OUT	DPZY	See Eq's (5.3.3-3)
YREC	y_r			(5.3.3-4)
ZREC	z_r			(5.3.3-5)
RHO	ζ_μ			(5.3.3-8)
SGR	$\sin\gamma_{r\ell}$			(5.3.3-7)
CGR	$\cos\gamma_{r\ell}$			(5.3.3-6)
SMULT				(5.3.3-2)
CMULT				(5.3.3-10)
L				Panel number in which the sending point 'j' lies
LS				Strip number in which the sending point 'j' lies
IO				See INCRO; Sec. 5.1.8
NBXS				
NCPNB				
IR				Index of sending point

Calling Subroutine GEND

Called Subroutine SUBP

Equations

When the receiving point 'i' lies on a z-oriented body, subroutine DPZY computes

$$DPZ_{ij} = \sum_{\mu=1}^{N\theta} DP(ARG_O, ARG_R, ARG_S) S(\theta_\mu, AR^{(b)}) \Delta\theta_\mu \qquad (5.3.3\text{-}1)$$

where

$$S(\theta_\mu, AR^{(b)}) = \frac{1}{\pi} \sin\theta_\mu \sqrt{1 + \cos^2\theta_\mu (AR^2-1)} \qquad (5.3.3\text{-}2)$$

$$\Delta\theta_\mu = \frac{\theta_{\mu+1} - \theta_{\mu-1}}{2} \qquad (5.3.3\text{-}3)$$

where

$$\theta_o = \theta_{N\theta} - 2\pi \text{ and } \theta_{N\theta+1} = \theta_1 + 2\pi$$

and $\mu = 1,2, \ldots N\theta$.

Note that $N\theta$ is either $N\theta^{(1)}$ or $N\theta^{(2)}$, depending on the body section in which the body receiving point lies; accordingly, θ_μ is either $\theta 1_\mu$ or $\theta 2_\mu$ - see input data, Sec. 1.2, and DP(ARG_0, ARG_R, ARG_R) is given by Eq.(5.5.3.7-2) computed in Subroutine SUBP.

Here, however, the receiving point arguments depend on the θ-distribution of the body section, and are defined as follows:

<u>ARG_R:</u>

$$x_i = XI_i$$

$$y_k = YB^{(b)} + a^{(b)} \cos\theta_\mu \qquad (5.3.3\text{-}4)$$

$$z_k = ZB^{(b)} + a^{(b)} AR^{(b)} \sin\theta_\mu \qquad (5.3.3\text{-}5)$$

$$\cos\gamma_{r_\ell} = \sin\theta_\mu/\zeta_\mu \qquad (5.3.3\text{-}6)$$

$$\sin\gamma_{r_\ell} = -AR^{(b)} \cos\theta_\mu/\zeta_\mu \qquad (5.3.3\text{-}7)$$

and

$$\zeta_\mu = \cos^2\theta_\mu(AR^2-1) + 1 \qquad (5.3.3\text{-}8)$$

A complete summary of the ARG_R and ARG_S are given in the table at the end of this section. ARG_0 denotes the constant arguments k_r and M - this holds true for all nine submatrices of the total downwash factor matrix [DT].

For receiving points on y-oriented bodies subroutine DPZY computes

$$DPY_{ij} = \sum_{\mu=1}^{N\theta} DP(ARG_0,ARG_R,ARG_S)C(\theta_\mu,AR^{(b)})\Delta\theta_\mu \qquad (5.3.3\text{-}9)$$

wher

$$C(\theta_\mu,AR^{(b)}) = \frac{1}{\pi}\cos\theta_\mu \sqrt{1 + \cos^2\theta_\mu(AR^2-1)} \qquad (5.3.3\text{-}10)$$

and

$\Delta\theta_\mu$ is given by Eq.(5.3.3-3)

DP(ARG_o, ARG_R, ARG_S) is given by Eq. (5.3.7-2) and the arguments ARG_R and ARG_S are summarized in the table below.

Receiving Point on Body, Sending Point on Panel

POINTS	ARGUMENTS	COORDINATES			DIHEDRAL ANGLE	θ	$\Delta\theta$	AR
		x	y	z				
Receiving point 'i'	ARG_{R_μ}	xI_i	y_μ	z_μ	γ_{rk}	θ_μ	$\Delta\theta_\mu$	$AR^{(b)}$
			See Eqs.(5.3.3-4 through -7)				See Eq. (5.3.3-3)	
						Sweep Angle	Box semi-width	Average box chord
	ARG_S		η_{c_ℓ}		γ_{s_ℓ}	λ_{s_j}		
Sending point 'j' strip 'ℓ'	$S(ARG_S)$		$-\eta_{c_\ell}$	ζ_{c_ℓ}	$-\gamma_{s_\ell}$	λ_{s_j}	e_ℓ	c_ℓ
	$G(ARG_S)$	ξ_{c_j}	η_{c_ℓ}					
	$S(G(ARG_S))$		$-\eta_{c_\ell}$		γ_{s_ℓ}	λ_{s_j}		

A detailed description of the "Arguments' above is given in Sec. 5.3.7 along with their usage in the computation of the DP-matrix element components.

5.3.4 SUBROUTINE DYPZ (KB, KS, LS, IZ, I, J1, J2, NYFLAG, FLND, FLNE, SGR, CGR, REFC, FMACH, KR, ARB, NBEA, LB0, LS0, JB0, DT)

Functional Description

Subroutine DYPZ prepares the variables necessary for the computation of one row of either the DYP-, or the DYZ-, or the DYY submatrix, depending on the location of the receiving point. In either case it calls subroutine SUBB in a do loop for each element of a row; latter is returned to the calling subroutine GEND via the argument list of DYPZ.

Input Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
KB				Body number in which receiving point 'i' lies
KS				Index of receiving point y- and z-coordinates
LS				See Subroutine DPZY Sec. 5.3.3
IZ				
I	i			
J1				Do-loop delimiters for the number of elements in one row of the submatrix
J2				
NYFLAG				0 for DYP and DYZ elements, 1 for DYY elements
FLND	δ	IN	ARG	See Blank Common Sec. 3.1
FLNE	ε			
SGR	$\sin\gamma_r$			
CGR	$\cos\gamma_r$			
REFC	$\bar{c}$			
FMACH	M			
KR	k_r			
ARB(10)	$AR^{(b)}$			
NBEA(10,2)				
LBO				Sequence number of first body with y-orientation

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
LSO		IN	ARG	y- and z-coordinate index for first y-oriented body element
JBO				Sending point index for first y-oriented body element
DT(500)		IN/OUT	ARG	One row of either of the submatrices DYP, DYZ or DYY
NDY		OUT	DYPZ	Flag used in subroutine SUBB - 1 for sending points in y-oriented bodies
LB				Body number in which sending point 'j' is located
JB				Sending point index (column no. of DT-matrix)
JZ				Body element number for sending body LB
NZYLB		IN		z-y orientation flag for body LB
SL CL TL SGS CGS		OUT		$\sin\lambda = 0$ $\cos\lambda = 1$ $\tan\lambda = 0$ $\sin\gamma_s = -1$ $\cos\gamma_s = 0$ Variables used in the called subroutine SUBB: $\lambda=0$ and $\gamma_s = -90^o$ for sending elements on y-oriented bodies

Calling Subroutine GEND

Called Subroutine SUBB

5.3.5 SUBROUTINE DZPY (KB, KS, LS, IZ, I, J1, J2, NYFLAG, FLND, FLNE, SGR, CGR, REFC, FMACH, KR, ARB, NBEA, DT)

Functional Description

Subroutine DZPY prepares the variables necessary for the computation

of one row of either the DZP-, or the DZZ-, or the DZY matrix, depending on the location of the receiving point. In either case it calls subroutine SUBB in a do loop for each element of a row; latter is returned to the calling subroutine GEND via the argument list of DZPY.

Input Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
KB				
KS				
LS				
IZ				
I	i			
J1				
J2				
NYFLAG				See Subroutine DYPZ
FLND	δ	IN	ARG	Sec. 5.3.4
FLNE	ε			
SGR	$\sin\gamma_r$			
CGR	$\cos\gamma_r$			
REFC	$\bar{c}$			
FMACH	M			
KR	k_r			
ARB(10)				
NBEA(10,2)				
DT(500)		IN/OUT		One row of either of the sub-matrices DZP, DZZ or DZY
NDY		OUT	DZPY	Flag used in subroutine SUBB; 0 for sending points in z-oriented bodies

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
LB		OUT	DZPY	See Subroutine DYPZ Sec. 5.3.4
JB				
JZ				
NZYLB		IN		
SL		OUT		$\sin\lambda=0$
CL				$\cos\lambda=1$
TL				$\tan\lambda=0$
SGS				$\sin\gamma_s=0$
CGS				$\cos\gamma_s=1$

Variables used in subroutine SUBB; $\lambda=0$, and $\gamma_s=0$ for sending elements on z-oriented bodies

<u>Calling Subroutine</u> GEND

<u>Called Subroutine</u> SUBB

5.3.6 SUBROUTINE SUBB (KB, KS, I, J, JZ; JB, LB, LS, NDY, NYFL, FLND, FLNE, PI, EPS, SGR, CGR, SGS, CGS, AR, SL, CL, TL, FL, BETA, SUM)

<u>Functional Description</u>

Subroutine SUBB computes the downwash factor matrix elements for all receiving points and sending points on interference bodies, one element at a time, according to Eqs. (5.3.6-1 and -2) or Eq. (5.3.6-3 and -4) depending on the orientation of the interference body in which the sending point is located. The actual computation of the components to the matrix elements is done in Subroutine DZY which is called from SUBB. The final result - one downwash factor, SUM - is returned to the calling subroutine GEND via the argument list of SUBB.

<u>Input Output Variables</u>

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
KB		IN	ARG	Index of receiving body - 0 when receiving point is on panel

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
KS				'Strip' number in which receiving point lies
I	i			Receiving point index - row number of DT-matrix
J	j			Sending point index - column number of DT-matrix
JZ				
JB				
LB				
LS				
NDY				
NYFL, NYFLAG		IN	ARG	See Subroutine DYPZ Sec. 5.3.4
FLND	δ			
FLNE	ε			
PI	π			
EPS	0.00001			
SGR	$\sin\gamma_r$			
CGR	$\cos\gamma_r$			
CGS	$\sin\gamma_s$			
CGS	$\cos\gamma_s$			
AR	AR.			
SL				$\sin\lambda = 0$
CL				$\cos\lambda = 1$
TL				$\tan\lambda = 0$
FL	$\bar{c}$			

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
BETA	β	IN	ARG	$\beta = \sqrt{1 - M^2}$
SUM	DT_{ij}	IN/OUT		One element of DT-matrix where the sending point 'j' is an interference body element
IND		OUT	SUBB	Flag for subroutine TKER; 0 for 'total' kernel, 1 otherwise
BR	$\bar{c}/2$			Reference semi-chord
ANOT				Local characteristic half-width of interference body element 'j'
DXS				Body element length
TEST1*		IN		$\lvert y_c^r - y_c^s \rvert$ — where y_c^r , z_c^r are the y- and z coordinates of receiving body centerline - and y_c^s, z_c^s are same for sending body centerline
TEST2*				$\lvert z_c^r - z_c^s \rvert$
D2D				See Eq. (5.3.6-2)
XX		OUT		See argument list of Sübroutine DZY; Sec. 5.1.3
Y				
Z				
XI1				
XI2				
ETA				
ZETA				
A0				
IDZDY				

* Used only if receiving point is an interference body element

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
IGO		IN	SUBB	Internal flag, 1 through 4, corresponding to the 'quadrant' in which the sending point lies: 1 - upper right (UR) 2 - upper left (UL) 3 - lower right (LR) 4 - lower left (LL)
DPUR				UR
DPUL				UL contribution to the
DPLR				LR downwash factor
DPLL				LL

Calling Subroutines DZPY and DYPZ

Called Subroutine and Common Blocks DZY, the Blank Common Block and the Labeled Common Block KDS

Equations

A. Sending points 'j' on z-oriented bodies ($\bar{b}$).

When the receiving points 'i' are on panels, subroutine SUBB computes

$$DZP_{ij} = DZ_{UR} + \delta DZ_{UL} + \varepsilon DZ_{LR} + \delta\varepsilon DZ_{LL} \qquad (5.3.6\text{-}1)$$

where the DZ(ARG_o, ARG_R, ARG_S) elements are described in Sec. 5.1.3 (Subroutine DZY).

When the receiving points i are on a z-oriented body 'b' subroutine SUBB computes DZZ_{ij} as follows:

a) If the two bodies b and $\bar{b}$ are not identical ($b \neq \bar{b}$) and their centerlines do not coincide,

$$DZZ_{ij} = DZ_{UR} + \delta DZ_{UL} + \varepsilon DZ_{LR} + \delta\varepsilon DZ_{LL}$$

b) if $b \neq \bar{b}$, but the centerlines of the two bodies coincide

$$DZZ_{ij} = 0$$

c) if $b = \bar{b}$, but $i \neq j$

$$DZZ_{ij} = 0$$

d) if $b = \bar{b}$ and $i = j$

$$DZZ_{ij} = D2D$$

$$= \frac{1.0}{2\pi(a_o^{\bar{b}})^2)1 + AR^{\bar{b}})} \qquad (5.3.6\text{-}2)$$

When the receiving point 'i' is on a y-oriented body, subroutine SUBB computes DZY_{ij} as follows:

a) if $b \neq \bar{b}$

$$DZY_{ij} = DZ_{UR} + \delta DZ_{UL} + \varepsilon DZ_{LR} + \delta\varepsilon DZ_{LL}$$

b) if $b = \bar{b}$

$$DZY_{ij} = 0$$

B. Sending points 'j' on y-oriented bodies ($\bar{b}$).

When the receiving points 'i' are on panels, subroutine SUBB computes

$$DYP_{ij} = DY_{UR} + \delta DY_{UL} + \varepsilon DY_{LR} + \delta\varepsilon D_{LL} \qquad (5.3.6\text{-}3)$$

where the DY(ARG_o, ARG_R, ARG_S) elements are described in Sec. 5.1.3.

When the receiving points are on z-oriented bodies

a) if $b \neq \bar{b}$

$$DYZ_{ij} = DY_{UR} + \delta DY_{UL} + \varepsilon DY_{LR} + \delta\varepsilon DY_{LL}$$

b) if $b = \bar{b}$, or $b \neq \bar{b}$ but their centerlines coincide

$$DYZ_{ij} = 0$$

When the receiving points are also on y-oriented bodies

a) if $b \neq \bar{b}$ and their centerlines do not coincide

$$DYY_{ij} = DY_{UR} + \delta DY_{UL} + \varepsilon DY_{OR} + \delta\varepsilon DY_{LL}$$

b) if $b \neq \bar{b}$, but their centerlines coincide

$$DYY_{ij} = 0$$

c) if $b = \bar{b}$, but $i \neq j$

$$DYY_{ij} = 0$$

d) if $b = \bar{b}$, and $i = j$

$$DYY_{ij} = DY2D = D2D/AR$$

$$= \frac{1}{AR} \quad \frac{1}{2\pi(a_o^{\bar{b}})^2 \ (1+AR^{\bar{b}})} \qquad (5.3.6\text{-}4)$$

The arguments that are used in the computations of the DZ and DY elements are tabulated below.

Receiving Point on Panel, Sending Point on Body

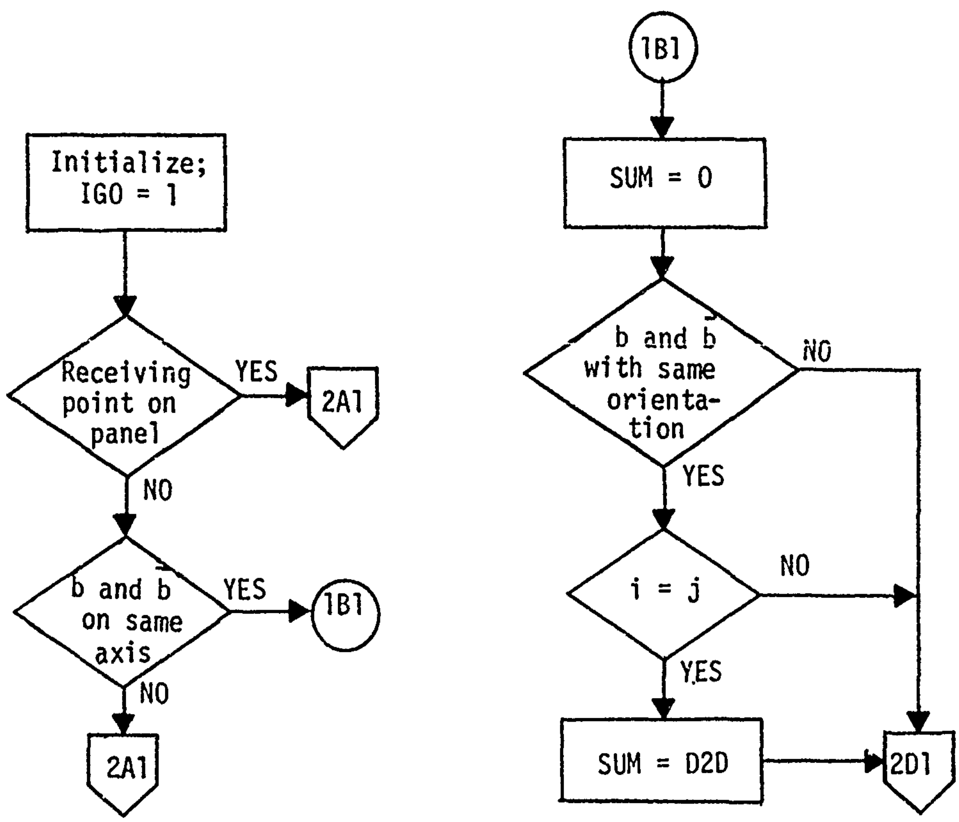
POINTS	ARGUMENTS	COORDINATES x	y	z	DIHEDRAL ANGLE	HALF WIDTH OF BODY	AVG RADIUS	AR
Receiving point i, strip k	ARG_R	x_i	y_k	z_k	γ_{Rk}			
Sending point j	ARG_S $S(ARG_S)$ $G(ARG_S)$ $S(G(ARG_S))$	ξ_{1_j} and ξ_{2_j}	η_c $-\eta_c$ η_c $-\eta_c$	ζ $-\zeta$	$\gamma_S=0$ for z-body $\gamma_S= -\pi/2$ for y-body	e^*	a_o	$AR^{(\bar{b})}$

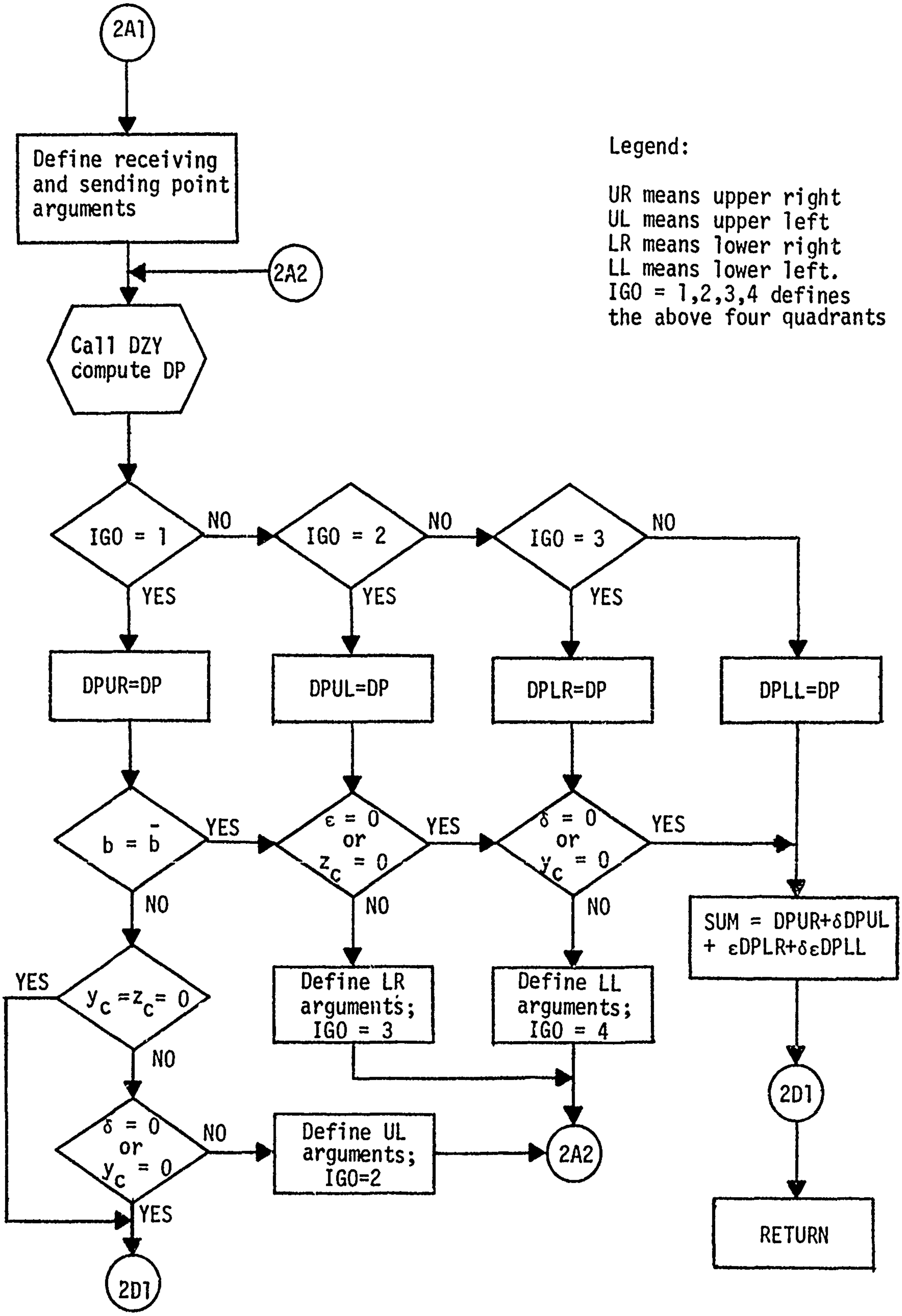
*Note that $e=a_o/2\sqrt{|1-AR|^2}$ for z-oriented, $AR\geq 1$ and for y-oriented $AR<1$ bodies and $e = a_o\sqrt{3}/2\sqrt{|1-AR|^2}$ for z-oriented, $AR<1$ and for y-oriented, $AR\geq 1$ bodies.

Receiving Point on Body, Sending Point on Body

POINTS	ARGUMENTS	COORDINATES x	y	z	DIHEDRAL ANGLE	Δx (OF BODY SECTION)	AVG RADIUS (OF BODY SECTION)	AR
Receiving point i	ARG_R	XI_i	$YB_k^{(b)}$	$ZB_k^{(b)}$	$\gamma_r=0$ for z-body, $\gamma_r = -\pi/2$ for y-body	Δx_k	a_o	$AR^{(b)}$
Sending point j	ARG_S	See Table above						

<u>Flow Chart - Subroutine SUBB</u>




2A1
Define receiving and sending point arguments
2A2
Call DZY compute DP
Legend:
UR means upper right
UL means upper left
LR means lower right
LL means lower left.
IGO = 1,2,3,4 defines the above four quadrants
IGO = 1
IGO = 2
IGO = 3
NO
YES
DPUR=DP
DPUL=DP
DPLR=DP
DPLL=DP
$b = \bar{b}$
$\varepsilon = 0$ or $z_c = 0$
$\delta = 0$ or $y_c = 0$
SUM = DPUR+δDPUL + εDPLR+$\delta\varepsilon$DPLL
Define LR arguments; IGO = 3
Define LL arguments; IGO = 4
$y_c = z_c = 0$
$\delta = 0$ or $y_c = 0$
Define UL arguments; IGO=2
2A2
2D1
RETURN
2D1

5.3.7 SUBROUTINE SUBP (I, L, LS, J, IO, JR, NBXS, NCPNB, SGR, CGR, YREC, ZREC, SUM, WORK)

Functional Description

Subroutine SUBP computes the downwash factor matrix elements, DP_{ij} for all receiving points on panels and interference bodies and sending points on panels, one element at a time, according to Equations (5.3.7-1 through -6). The computation of the individual components that make up each downwash factor element is done in Subroutines SNPDF and INCRO, which are called from SUBP. Additional arguments that are needed for the calculation of the downwash factor contribution of image sending points inside associated bodies are calculated by subroutine SUBI, which is also called from SUBP. The resulting total downwash factor element 'SUM' is returned to the calling subroutine via the argument list of SUBP.

Input Output Variables

<table>
<tr><th>MNEMONIC</th><th>SYMBOL</th><th>IN/OUT</th><th>SOURCE</th><th>DESCRIPTION</th></tr>
<tr><td>I
L
LS</td><td></td><td rowspan="4">IN</td><td rowspan="4">ARG</td><td>See DPPS, Sec. 5.3.2</td></tr>
<tr><td>J</td><td>j</td><td>Running index of the element number in DPP matrix row (column number)</td></tr>
<tr><td>IO
IR
NBXS
NCPNB</td><td></td><td>See INCRO, Sec. 5.1.7</td></tr>
<tr><td>SGR
CGR
YREC
ZREC</td><td>$\sin\gamma_r$
$\cos\gamma_r$
y_r
z_r</td><td>See DPPS, Sec. 5.3.2</td></tr>
</table>

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
SUM		IN/OUT		One element of the DT matrix for receiving point 'i', sending point 'j', when 'i' is on a panel. Note that, if 'i' represents a receiving interference body section, SUM is only the contribution (to this DT-element) of one point on the surface of the body section – see Eq.(5.3.3-1).
TL SL CL XO YO ZO ES CV				See Subroutine SNPDF, Sec.5.1.11
AX AY AZ AX1 AY1 AZ1 AX2 AY2 AZ2		OUT	SUBP	See Subroutine INCRO, Sec. 5.1.8
DIJS DIJI	D I(D)	IN	Argument List of SNPDF	Steady contribution to the downwash factor: of sending point 'j' on panel (DIJS); of the image of 'j' inside the current associated body (DIJI)
WORK		IN/OUT	ARG	Complex work array

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
DELRS	Re(ΔD)	IN	Argument List of INCRO	Real part of the unsteady contribution to the downwash factor: of sending point 'j' on panel
DELIS	Re[I(ΔD)]			Real part of the unsteady contribution to the downwash factor: of the image of 'j'
DELRI	Im(ΔD)			Imaginary part of unsteady contribution to the downwash factor
DELII	Im[I(ΔD)]			
DPUR		IN	SUBP	See Eqs. (5.3.7-2 through -6)
DPUL				
DPLR				
DPLL				
NOAS				The number of associated bodies for panel in which the sending point 'j' lies
NA1				Delimiters to the do-loop for all associated bodies of panel
NA2				
DA		OUT		Arguments to Subroutine SUBI; see Sec. 5.1.12
DZB				
DYB				
DAR				
DETA				
DZETA				
DCGAM				
DSGAM				
DEE				
DXI		IN	Argument List of SUBI	Image point arguments; see Sec. 5.1.12
DETAI				
DZETAI				
DCGAMI				
DSGAMI				
DEEI				
DTLAMI				

$I(ARG_S)$* = images of ARG_S inside associated body IR
$S(ARG_S)$ = images of ARG_S with respect to the $y = 0$ plane
$G(ARG_S)$ = images of ARG_S with respect to the $z = 0$ plane

The arguments that constitute ARG_R and ARG_S are tabulated below.

Receiving Point on Panel, Sending Point on Panel

Points	Arguments	Coordinates			Dihedral Angle	Sweep Angle	Box Half Width	Arg Box Chord
		x	y	z				
Receiving point 'i', strip 'k'	ARG_R	x_i	y_k	z_k	γ_{rk}			
Sending point 'j', strip 'ℓ'	ARG_S	ξ_{c_j}	η_{c_ℓ}	ζ_{c_ℓ}	γ_{s_ℓ}	λ_{s_j}	e_ℓ	c_ℓ
	$S(ARG_S)$		$-\eta_{c_\ell}$		$-\gamma_{s_\ell}$	$-\lambda_{s_j}$		
	$G(ARG_S)$		η_{c_ℓ}	$-\zeta_{c_\ell}$				
	$S[G(ARG_S)]$		$-\eta_{c_\ell}$		γ_{s_ℓ}	λ_{s_j}		

For receiving points on interference body elements, the ARG_R are defined in Section 5.5.3.3 (Subroutine DPZY).

The evaluation of D in Equations (5.3.7-3 through -6) is done by the following subroutines:

SNPDF — for the steady case ($k_r = 0$), yields $D^{(s)}$;

SNPDF, INCRO, TKER, IDF1 and IDF2 — for the unsteady case, yielding ΔD ($\Delta D = 0$ for steady cases).

Then, in general

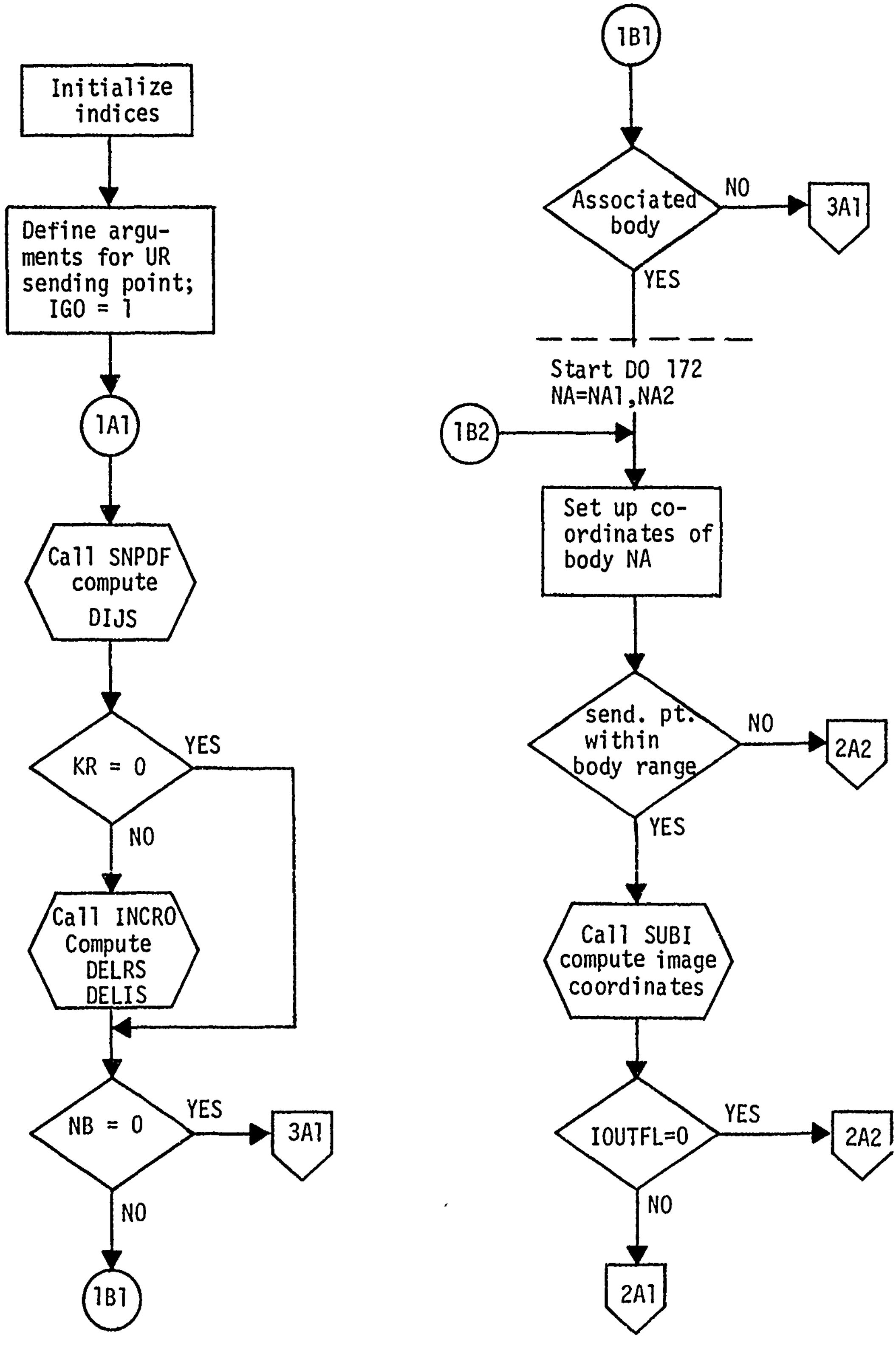
$$D = D^{(s)} + \Delta D \tag{5.3.7-7}$$

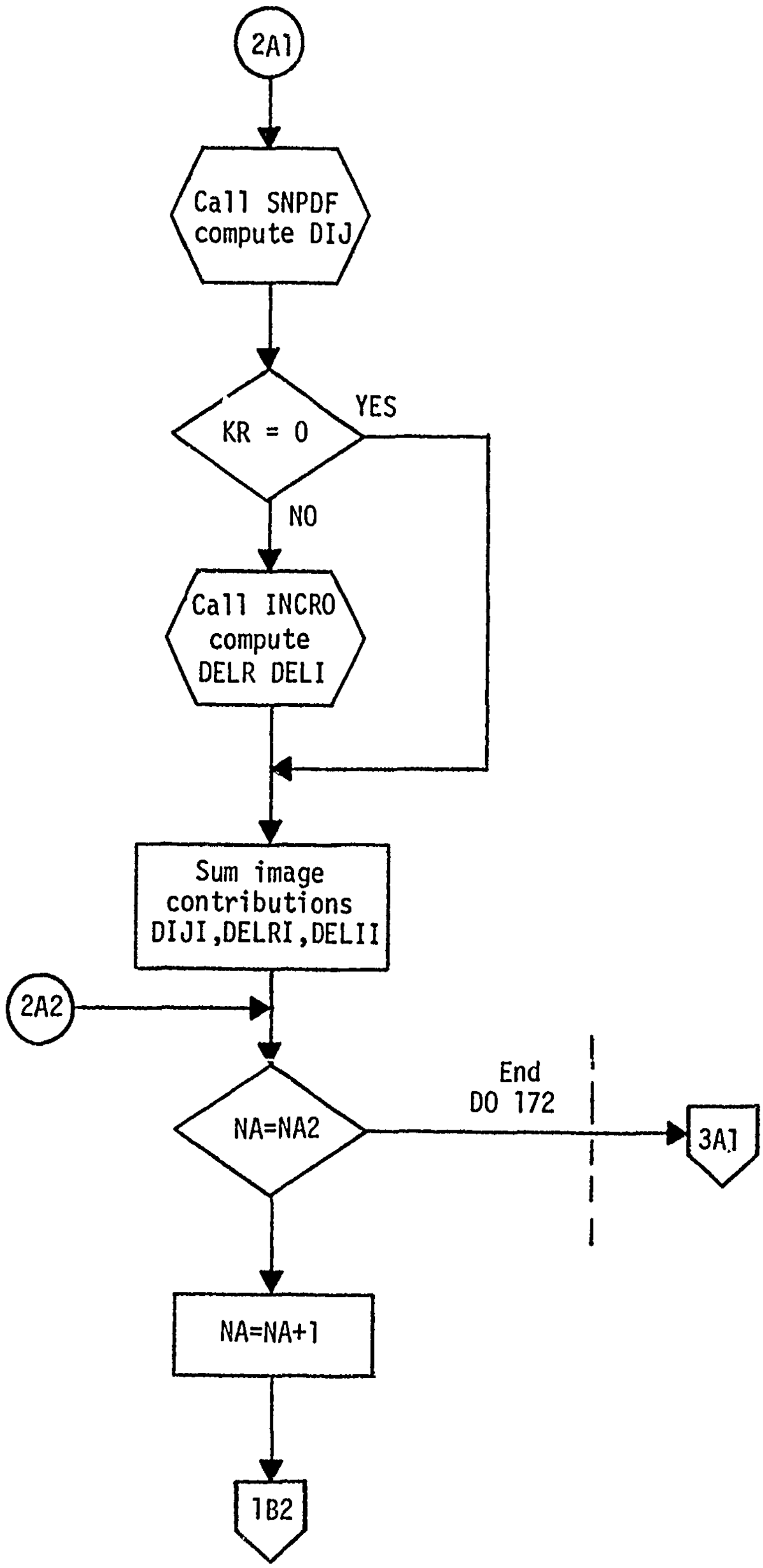
Description of the above five subroutines is given in Section 5.1.

---

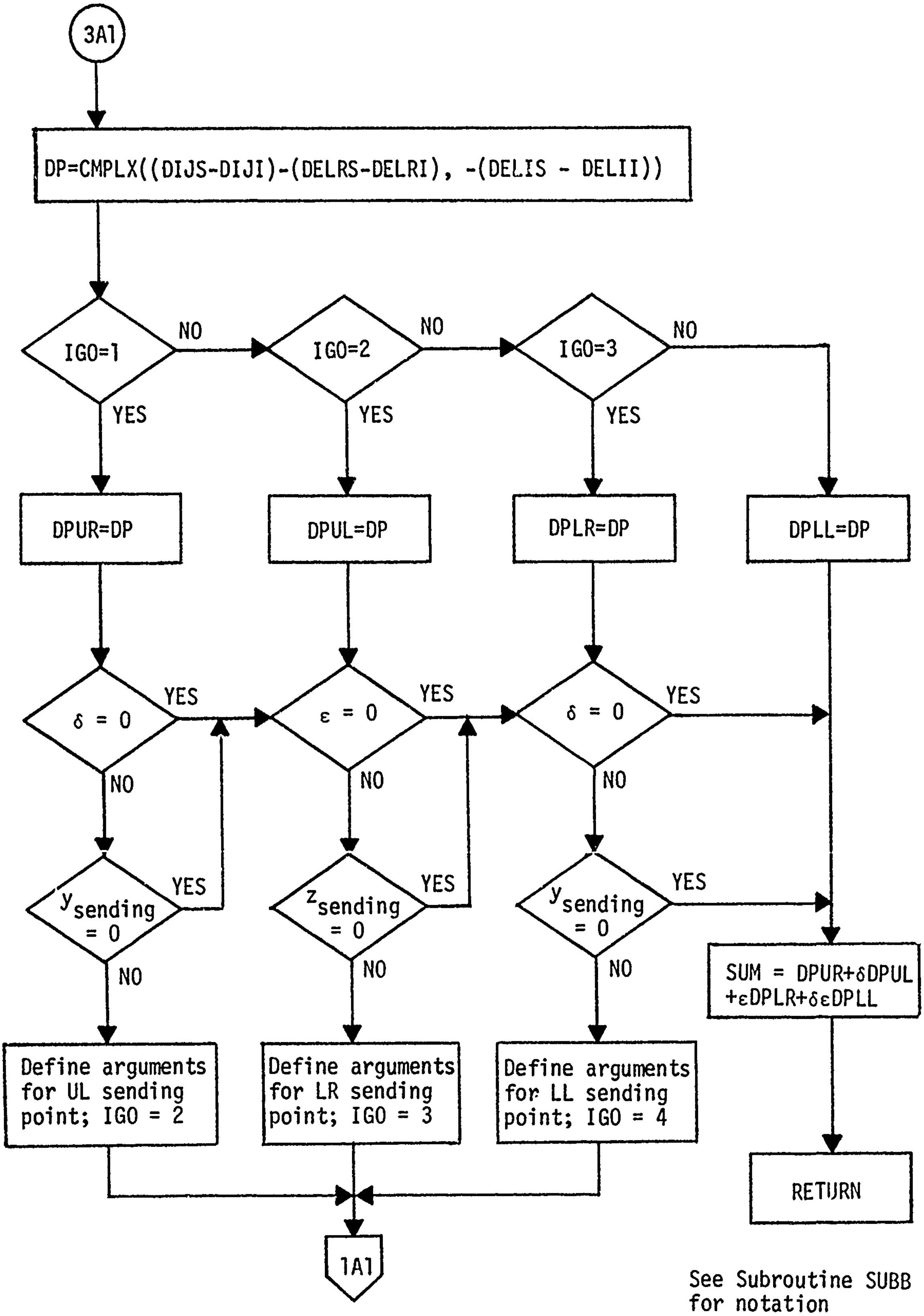
*$I(ARG_S)$ is computed by Subroutine SUBI described in Sec. 5.1.

Flow Chart - Subroutine SUBP





2A1
Call SNPDF
compute DIJ
KR = 0
YES
NO
Call INCRO
compute
DELR DELI
Sum image
contributions
DIJI,DELRI,DELII
2A2
NA=NA2
End
DO 172
3A1
NA=NA+1
1B2



See Subroutine SUBB for notation

Calling Subroutines   DPPS and DPZY

Called Subroutines and Common Blocks Subroutines SNPDF, INCRO and SUBI, and the Blank Common Block.

Equations

Subroutine SUBP computes one element of either the DPP or the DPZ or the DPY-matrix, depending on the location of the receiving points.

$$\left.\begin{array}{l} DPP_{ij} \\ DPZ_{i_\mu j} \\ DPY_{i_\mu j} \end{array}\right\} = DP(ARG_0,\ ARG_R,\ ARG_S) \qquad (5.3.7\text{-}1)$$

for receiving point 'i' on panels, or 'i_μ' on bodies and sending point 'j' on panels, where

$$DP = DP_{UR} + \delta DP_{UL} + \varepsilon DP_{LR} + \delta\varepsilon DP_{LL} \qquad (5.3.7\text{-}2)$$

and

$$DP_{UR} = D(ARG_S) - \sum_{IR=RANGE^{(p)}} D[I(ARG_S)] \qquad (5.3.7\text{-}3)$$

$$DP_{UL} = D[S(ARG_S)] - \sum_{IR=RANGE^{(p)}} D\{I[S(ARG_S)]\} \qquad (5.3.7\text{-}4)$$

$$DP_{LR} = D[G(ARG_S)] - \sum_{IR=RANGE^{(p)}} D\{G[S(ARG_S)]\} \qquad (5.3.7\text{-}5)$$

$$DP_{LL} = D\{S[G(ARG_S)]\} - \sum_{IR=RANGE^{(p)}} D\Big(I\{S[G(ARG_S)]\}\Big) \qquad (5.3.7\text{-}6)$$

where

ARG_0 represents the constant arguments k_r and M;
ARG_R represents the variable receiving point arguments
ARG_S = sending point arguments

and

$RANGE^{(p)}$ refers to the bodies associated with panel p in which the sending point lies

5.4 Segment 4

5.4.1 SUBROUTINE RDMODE(IA, NA, NIN, NM, NOUT)

Functional Description

This routine reads the panel and body modal data from cards. The data for each modal coefficient consists of the coefficient and an integer containing the mode number, the panel or body number, the exponent for the x coordinate, and a flag indicating whether relative or absolute coordinates for x and y are to be used. This data is then sorted according to mode number, then panel or body number and then the x and y exponents, respectively. The routine returns the input coefficients and their identifiers, the number of coefficients for panels and the z and y oriented bodies and the number of modes.

Input-Output Variables

MNEMONICS	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
IA		OUT	ARG	Array containing the modal coefficients and coefficient description for panels and bodies.
NA		OUT	ARG	Array containing the number of coefficients for panels and z and y oriented bodies.
NIN		IN	ARG	I/O unit number containing the input modal data.
NM		OUT	ARG	Number of modes
NOUT		IN	ARG	I/O unit number for printing on.

Calling Subroutines MAIN

Error Messages

SUBROUTINE **RDMODE** INVALID DATA CARD. -- IGNORED --

The data card which has been read is invalid and has been ignored.

5.5 Segment 5

5.5.1 SUBROUTINE SB (A, NM, NAY, NAZ, WORK, NWORK)

Functional Description

Subroutine SB, 'Slender Body', calculates the normalwash at lifting surface boxes and interference body elements due to slender body elements.

Input Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
A		IN	ARG	Array containing the modal data read by subroutine RDMODE.
X		IN	Blank Common Block	Array of box and interference body elements x-coordinate.
AR	AR	IN	Blank Common Block	Array of body aspect ratios.
AO	a_0	IN	Blank Common Block	Array of body radius.
KR	k_r	IN	Blank Common Block	Reduced frequency.
NB	NB	IN	Blank Common Block	Number of bodies.
NM	NM	IN	ARG	Number of modes.
AOP	a_0'	IN	Blank Common Block	Array of rate of change of body radius with rexpect to x.
NAY		IN	ARG	Number of modal coefficients for y-oriented bodies.
NAZ		IN	ARG	Number of modal coefficients for z-oriented bodies.
NBY	NBY	IN	Blank Common Block	Number of y-oriented bodies.
NBZ	NBZ	IN	Blank Common Block	Number of z-oriented bodies.
NTP	NTP	IN	Blank Common Block	Number of lifting surface boxes.
NTY	NTY	IN	Blank Common Block	Number of y-oriented interference body elements.
NTZ	NTZ	IN	Blank Common Block	Number of z-oriented interference body elements.
CBAR	$\overline{c}$	IN	Blank Common Block	Reference chord length.
NSBE	NSBE	IN	Blank Common Block	Array of number of slender body elements per body.
NTYS	NTYS	IN	Blank Common Block	Number of y-oriented slender body elements.

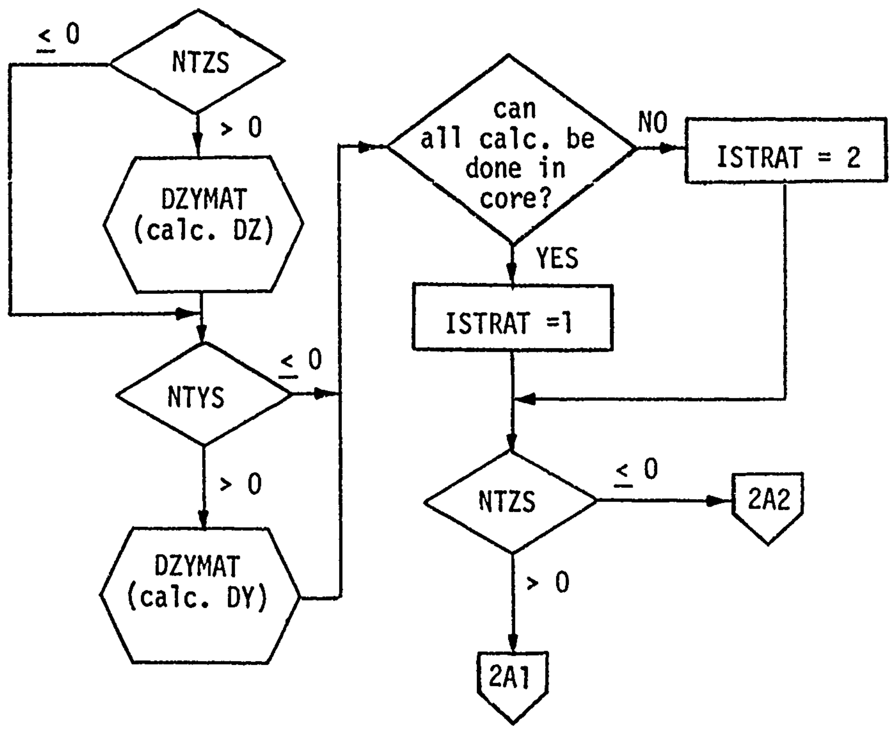
MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
NTZS	NTZS			Number of z-oriented slender body elements
WORK	S			Complex working array
XIS1	ξ_{s1}	IN	ARG	Array containing the leading edge x-coordinate of the slender body elements
XIS2	ξ_{s_2}			Array containing the trailing edge x-coordinate of the slender body elements
FMACH	M			Mach number
NWORK				Length of working array WORK

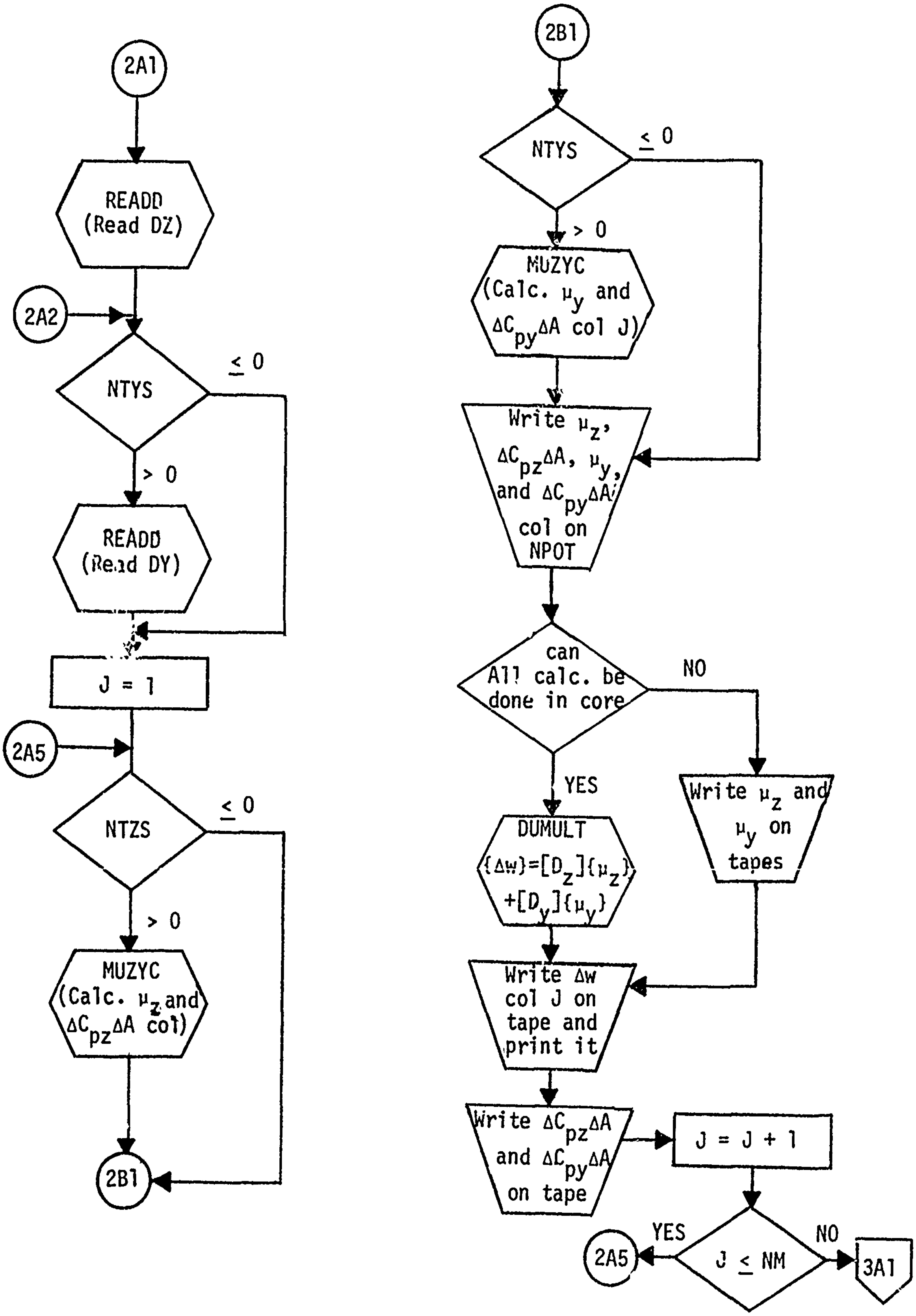
<u>Calling Subroutines</u> MAIN

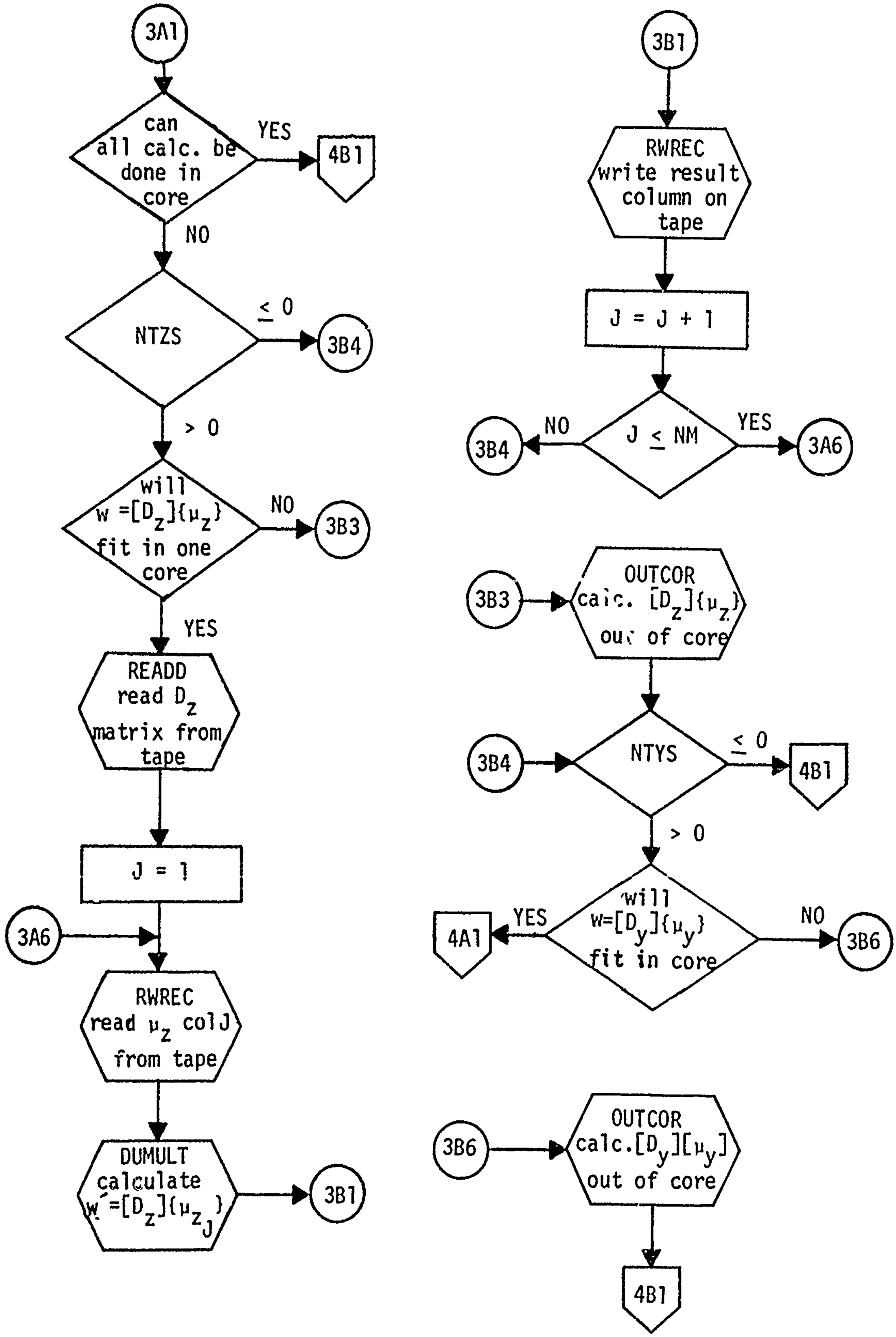
<u>Called Subroutines and Common Blocks</u>

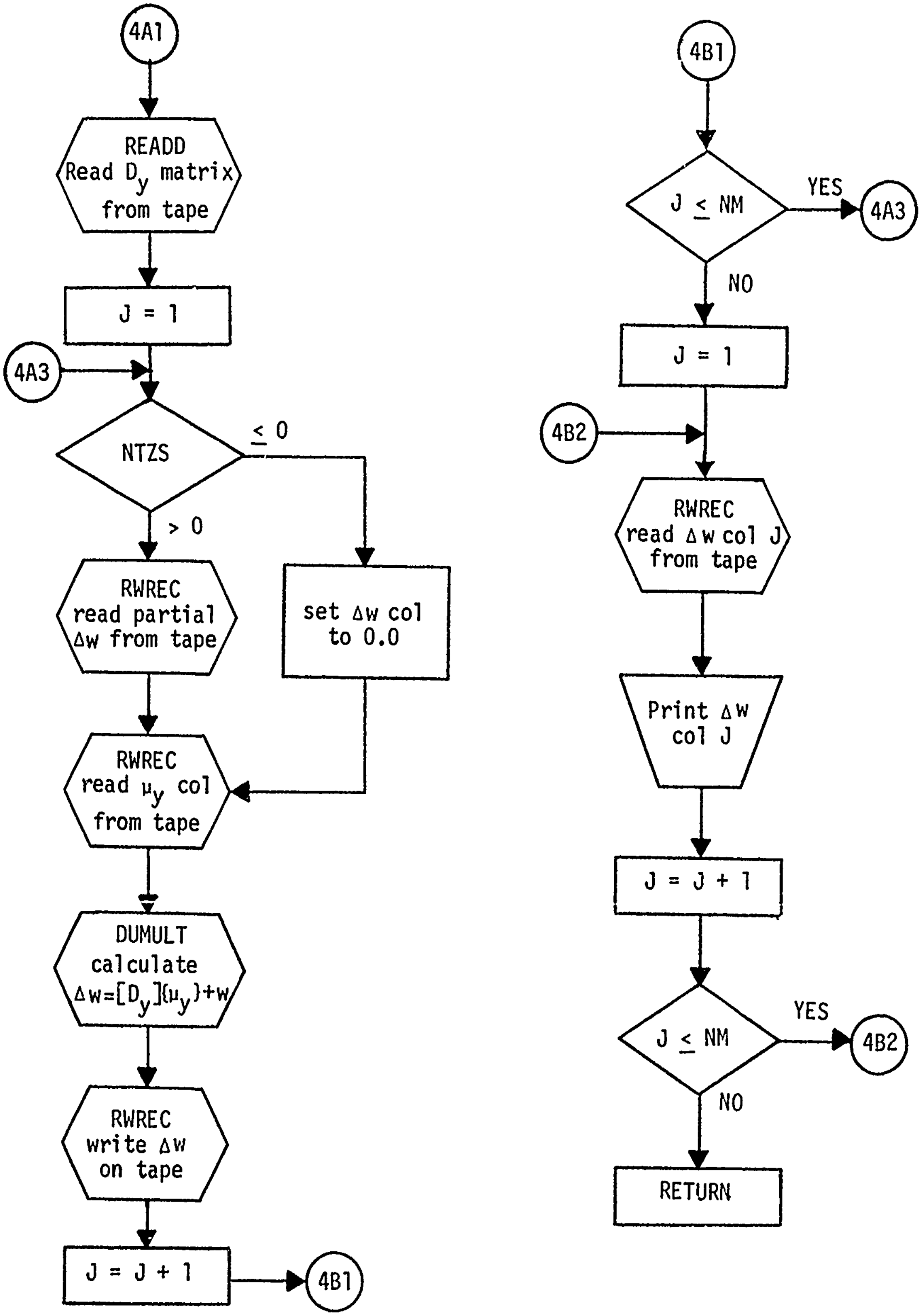
Blank Common, RWREC, MYZYC, DUMULT, READD, DZYMAT, OUTCOR

<u>Flow Chart - Subroutine SB</u>




2A1
READD (Read DZ)
2A2
NTYS
≤ 0
> 0
READD (Read DY)
J = 1
2A5
NTZS
≤ 0
> 0
MUZYC (Calc. μ_z and $\Delta C_{pz}\Delta A$ col)
2B1
2B1
NTYS
≤ 0
> 0
MUZYC (Calc. μ_y and $\Delta C_{py}\Delta A$ col J)
Write μ_z, $\Delta C_{pz}\Delta A$, μ_y, and $\Delta C_{py}\Delta A$ col on NPOT
can All calc. be done in core
NO
YES
Write μ_z and μ_y on tapes
DUMULT $\{\Delta w\}=[D_z]\{\mu_z\}+[D_y]\{\mu_y\}$
Write Δw col J on tape and print it
Write $\Delta C_{pz}\Delta A$ and $\Delta C_{py}\Delta A$ on tape
J = J + 1
J ≤ NM
YES
NO
2A5
3A1


3A1
can all calc. be done in core
YES
4B1
NO
NTZS
≤ 0
3B4
> 0
will w =[Dz]{μz} fit in one core
NO
3B3
YES
READD read Dz matrix from tape
J = 1
3A6
RWREC read μz colJ from tape
DUMULT calculate w´=[Dz]{μzJ}
3B1
3B1
RWREC write result column on tape
J = J + 1
J ≤ NM
NO
3B4
YES
3A6
3B3
OUTCOR calc. [Dz]{μz} out of core
3B4
NTYS
≤ 0
4B1
> 0
will w=[Dy]{μy} fit in core
YES
4A1
NO
3B6
3B6
OUTCOR calc.[Dy][μy] out of core
4B1


4A1
READD
Read D_y matrix
from tape
J = 1
4A3
NTZS
≤ 0
> 0
RWREC
read partial
Δw from tape
set Δw col
to 0.0
RWREC
read μ_y col
from tape
DUMULT
calculate
$\Delta w=[D_y]\{\mu_y\}+w$
RWREC
write Δw
on tape
J = J + 1
4B1
4B1
J ≤ NM
YES
4A3
NO
J = 1
4B2
RWREC
read Δw col J
from tape
Print Δw
col J
J = J + 1
J ≤ NM
YES
4B2
NO
RETURN

5.5.2 SUBROUTINE DUMULT (N1, N2, NTZS, NTYS, W, DZ, UZ, DY, UY)

Functional Description

This subroutine performs the following complex matrix operation.

$$\left\{\Delta W_{OUT}\right\} = \left[D_z\right] \left\{\mu_z\right\} + \left[D_y\right] \left\{\mu_y\right\} + \left\{\Delta W_{IN}\right\}$$

The result of this operation (Δw) is returned to the calling routine as w.

Input Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
DY	D_y			Input matrix.
DZ	D_z			Input matrix
N1				The first row of Δw to calculate
N2				The last row of Δw to calculate
NTYS		IN	ARG	Number of columns in D_y
NTZS				Number of columns in D_z
UY	μ_y			Input matrix
UZ	μ_z			Input matrix
W	Δw	IN/OUT	ARG	Input matrix and output result

Calling Subroutines SB, OUTCOR

5.5.3 SUBROUTINE DZYMAT (D, NFB, NLB, NTZYS, IDZDY, NTAPE, XP, BETA)

Functional Description

This subroutine sets up the proper argument lists for the calculation of each row of the D_z or D_y matrix and then calls subroutine ROWDYZ.

Input Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
D		IN	ARG	Working array used to store a row of D_z or D_y
CG	$\cos\gamma$			Array containing the cosine of the lifting surface strip dihedral angle

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
NB	NB			Total number of bodies
NC	NC			Array containing the number of chordwise boxes per strip in a panel
NP	NP			Number of panels
NS	NS			Array containing the number of strips in a panel
SG	$\sin\gamma$			Array containing the sine of the lifting surface strip dihedral angle
XP		IN	ARG	x-control point coordinate of lifting surface boxes
YB				y-coordinate of center of bodies
YP				y-control point coordinate of lifting surface strip
ZB				z-coordinate of center of bodies
ZP				z-control point coordinate of lifting surface strip
NBY				Number of y oriented bodies
NBZ				Number of z oriented bodies
NFB				Number of the first body with the orientation requested
NLB				Number of the last body with the orientation requested
NTP				Total number of boxes
BETA	β	IN	ARG	$\sqrt{1 - M^2}$
MACH	M			Mach number
NBEA				Array containing number of interference body elements per body and the body orientation
IDZDY				Flag indicating whether the $D_z(o)$ or $D_y(1)$ matrix is to be calculated
NTAPE				I/O unit number which the output matrix is to be written on
NTZYS				Number of z or y oriented slender body elements

Calling Subroutines SB

Called Subroutines and Common Blocks
ROWDYZ, Blank Common Block

5.5.4 SUBROUTINE MUZYC (NMODE, NCOEF, K, NTZY, NFBODY, NLBODY, NSBE, KR, IA, A, CBAR, AO, AOP, XIS1, XIS2, AR, UZY, CPZY)

Functional Description

Subroutine MUZYC calculates the axial doublet strengths and loading for slender bodies.

Input Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
A				Real array containing the modal coefficients and a key identifying the coefficients.
K				Code identifying which type coefficient to use = 2 use a_z = 3 use a_y
AR	AR	IN	ARG	Array containing aspect ratios of the bodies.
AO	a_o			Array containing the radii of the bodies.
IA				Integer array equivalent to the array "A".
KR	k_r			Reduced frequency.
AOP	a_o'			
UZY	$\bar{\mu}_z, \bar{\mu}_y$	OUT		The μ_z or μ_y column
CBAR	$\bar{c}$	IN		Reference chord length.
CPZY	$\Delta C_{p_z} \Delta A(1+AR)$	OUT	ARG	
	$\Delta C_{p_y} \Delta A(1+AR)$	OUT		
NSBE		IN		Array containing the number of slender body elements per body
NTZY	NTZS NTYS	IN	ARG	Total number of z or y oriented slender body elements
XIS1	ξ_{s1}	IN		Array containing the leading edge x-coordinate of the slender body elements.

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
XIS2	ξ_{s2}			Array containing the trailing edge x-coordinate of the slender body elements
NCOEF				Number of modal coefficients input for this orientation of body.
NMODE		IN	ARG	Number of modes in analysis.
NFBODY				Number of the first body with orientation requested.
NLBODY				Number of the last body with orientation requested.

Calling Subroutines SB

Equations

$$\bar{\mu}_z \text{ and } \bar{\mu}_y = \frac{ws}{D2D}$$

$$\left.\begin{array}{c} \Delta C_{p_z} \Delta A \ AR \\ \text{and} \\ \Delta C_{p_y} \Delta A \ AR \end{array}\right\} = \Big[(ws)L + i(ws')\,(M)\Big]\; 2a_o \ AR$$

where for

$\underline{K = 2}$ or z subscript

$$D2D = \frac{1.0}{2\pi\,(1 + AR)\,a_o^2}$$

$$L = 2\pi\,(a_o' + i\,k_r/\bar{c})$$

$$M = \pi\,a_o/\bar{c}$$

K = 3 or y subscript

$$D2D = \frac{1.0}{2\pi\ (1 + AR)\ a_0^{\,2}\ AR}$$

$$L = 2\pi\ AR\,(a_0' + i\ k_r/\ \bar{c})$$

$$M = AR\,\pi\ a_0/\ \bar{c}$$

and

$$ws = wsR + i\ wsI$$

$$wsR = \sum_{n=o}^{9} A\ n \left(\frac{xs}{\bar{c}}\right)^{n-1}$$

$$wsI = \sum_{n=1}^{9} A\ 2k_r \left(\frac{xs}{\bar{c}}\right)^{n}$$

$$ws' = wsR' + i\ wsR(2k_r)$$

$$wsR' = \sum_{n=1}^{9} A\ n(n-1) \left(\frac{xs}{\bar{c}}\right)^{n-2}$$

5.5.5 SUBROUTINE OUTCOR (WORK, NWORK, NTS, N, NM, NUTAP, NDWIN, NDWOUT, NDTAP)

Functional Description

This routine performs the following matrix operation

$$[\Delta w] = [\Delta w'] + [D]\ [U]$$

It is assumed that all the matrices will not fit into core. As many columns of U and $\Delta w'$ are read into core as possible and the matrix D is read a row at a time performing the necessary operations to calculate the columns of Δw which are written on an I/O unit. This operation is repeated until all the columns of U have been read and hence all of Δw written.

Input Output Variables

MNEMONICS	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
WORK				Complex working array.
NWORK				Length of working array.
NTS				Number of columns in the matrix D.
N				Number of rows in the matrix D.
NM				Number of columns in the matrix U.
NUTAP				I/O unit number containing the matrix U in column sort.
NDWIN		IN	ARG	I/O unit number containing the $\Delta w'$ matrix in column sort. If it is equal 0, the w' matrix is set to zero.
NDWOUT				I/O unit number which the Δw matrix is written on in column sort.
NDTAP				I/O unit number containing the D matrix in row sort.

Calling Subroutines SB

Called Subroutines and Common Blocks

RWREC, DUMULT, READD

Error Messages

SUBROUTINE **OUTCOR** NUMBER OF MODES IS LESS THAN OR EQUAL TO ZERO.
CALCULATIONS SKIPPED

5.5.6 SUBROUTINE ROWDYZ (NFB, NLB, ROW, NTZYS, D, DX, DY, DZ, BETA, IDZDY, NTAPE, SGR, CGR)

Functional Description

This routine performs the logic required to set up the argument list to DZY for the purpose of calculating a row of the D_z or D_y matrix.

Input Output Variables

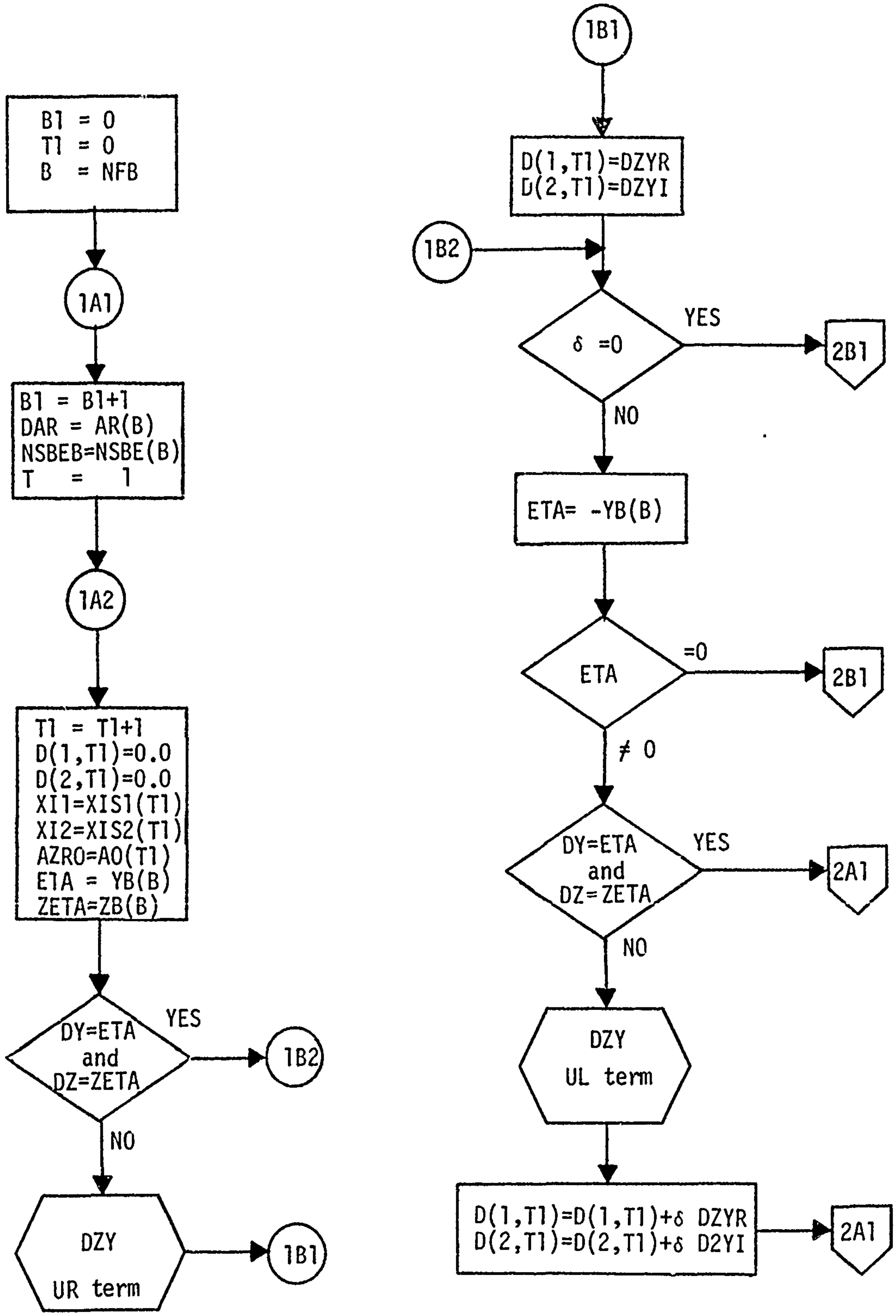
MNEMONICS	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
D		OUT	ARG	The output row of D_z or D_y (complex)
AR	$\mathcal{R}$	IN	Blank Common Block	Aspect ratio at the body.
AO	a_o	IN	Blank Common Block	Radius of the body.
DX	X	IN	ARG	x-coordinate of receiving point.

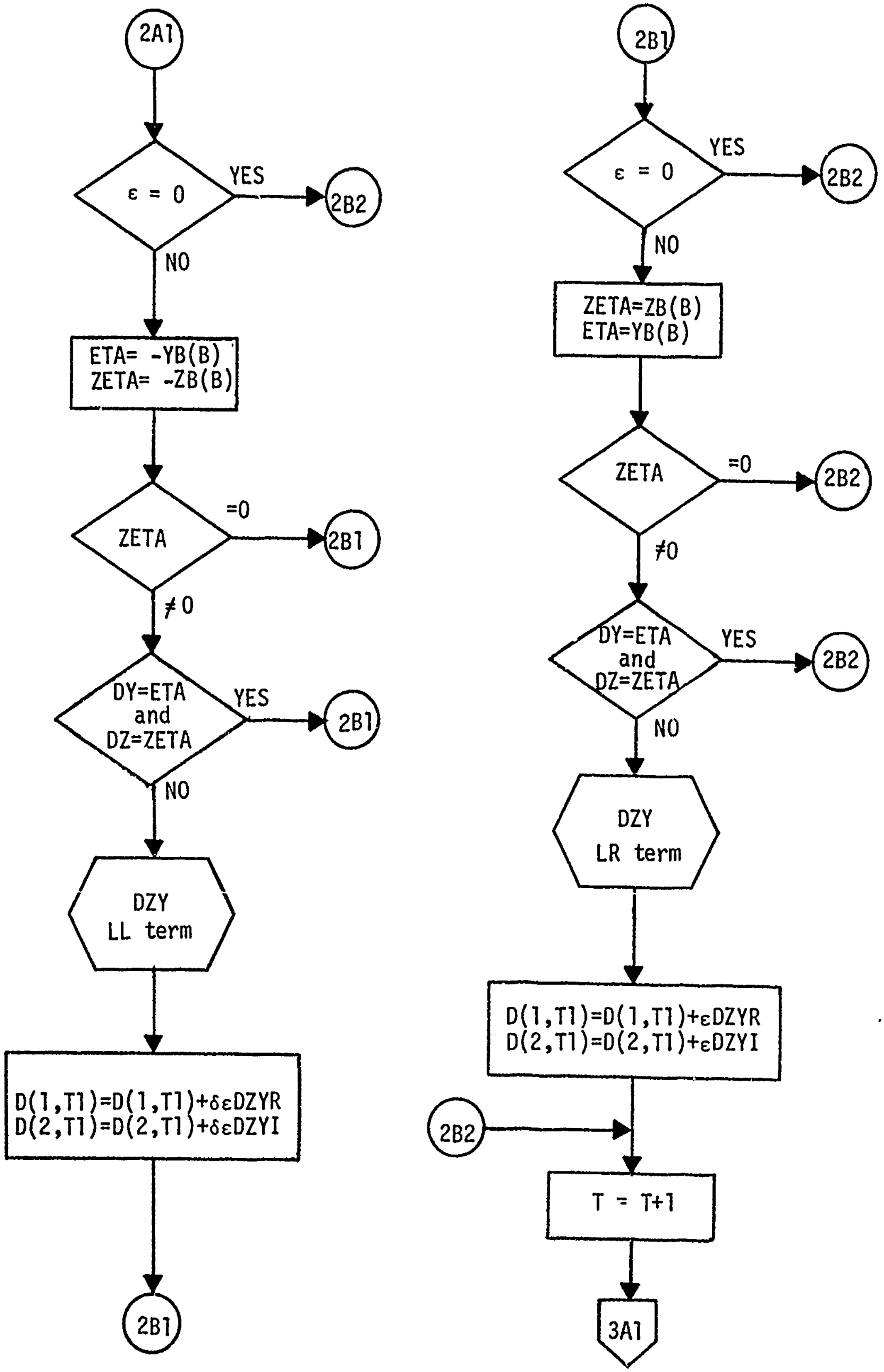
MNEMONICS	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
DY	Y	IN	ARG	y-coordinate of receiving point.
DZ	Z	IN	ARG	z-coordinate of receiving point.
KR	k_r	IN	Blank Common Block	Reduced frequency.
ND	δ			Symmetry flag.
NE	ε			Ground effects flag.
YB	y_B			Array containing y-coordinates of bodies
ZB	z_B			Array containing z-coordinates of bodies.
CGR	$\cos\gamma_r$	IN	ARG	Cosine of dihedral angle of receiving point.
NFB				Number of the first body having the desired z or y orientation.
NLB				Number of the last body having the desired z or y orientation.
ROW				The row number of D_z or D_y to be calculated.
SGR	$\sin\gamma_r$			Sine of dihedral angle of receiving point.
BETA	β			$\sqrt{1 - M^2}$
CBAR	c	IN	Blank Common Block	Length of reference chord
MACH	M			Mach number.
NSBE				Array containing the number of slender body elements per body
XIS1	ξ_{s1}			Array containing the x-coordinate of the leading edges of the slender body elements
XIS2	ξ_{s2}			Array containing the x-coordinate of the training edges of the slender body elements

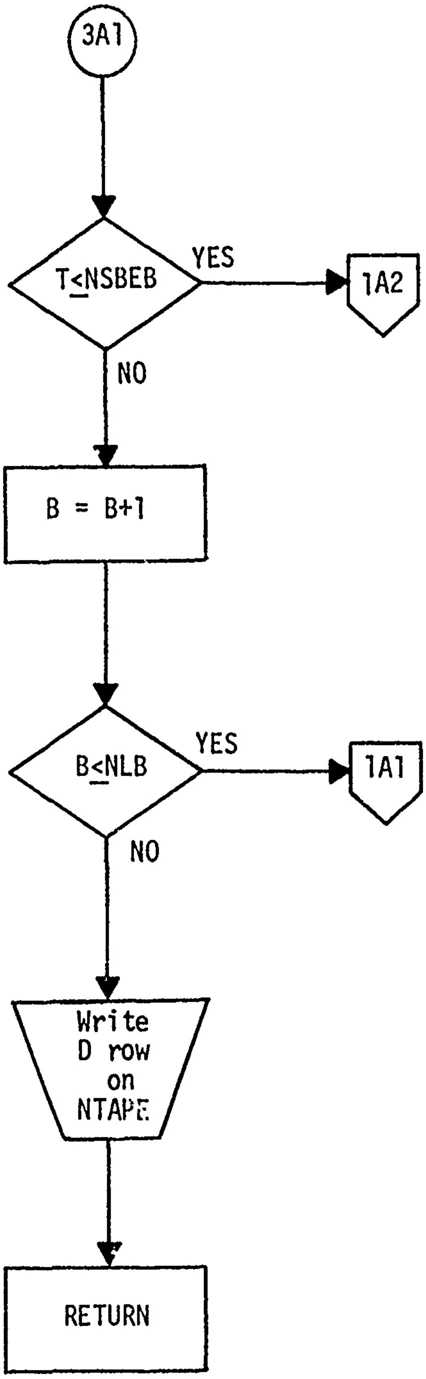
Calling Subroutines DZYMAT

Called Subroutines and Common Blocks Subroutine DZY and the Blank Common Block

Flow Chart - Subroutine ROWDYZ




2A1
ε = 0
YES
2B2
NO
ETA= -YB(B)
ZETA= -ZB(B)
ZETA
=0
2B1
≠0
DY=ETA
and
DZ=ZETA
YES
2B1
NO
DZY
LL term
D(1,T1)=D(1,T1)+δεDZYR
D(2,T1)=D(2,T1)+δεDZYI
2B1
2B1
ε = 0
YES
2B2
NO
ZETA=ZB(B)
ETA=YB(B)
ZETA
=0
2B2
≠0
DY=ETA
and
DZ=ZETA
YES
2B2
NO
DZY
LR term
D(1,T1)=D(1,T1)+εDZYR
D(2,T1)=D(2,T1)+εDZYI
2B2
T = T+1
3A1



See Subroutine SUBB
for notation

5.6 Segment 6

5.6.1 SUBROUTINE WANDWT (A, IA, NSARRY, NBARAY, X, YP, Y1, ZP, Z1, NR, CBAR, KR, NP, NTP, NM, COEF, NDW, NWT, NOUT, W, DW, NB, IPRINT)

Functional Description

This subroutine calculates the complex boundary conditions (w) on the lifting surfaces due to lifting surface motion and adds to it the incremental normalwash, Δw, due to the slender bodies. Also included in Δw is a normalwash induced at the interference body elements.

Input Output Variables

MNEMONICS	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
A	a	IN	ARG	Array containing modal coefficients
W	w	OUT	ARG	A column of the matrix w
X	xp	IN	ARG	3/4 chord x-coordinate t lifting surface box
DW	Δw	IN	I/O(NDW)	Input Δw matrix column
	WT	OUT	I/O(NWT)	The output matrix column WT
IA				Coded array describing the modal coefficients
KR	k_r			Reduced frequency
NM	NM			Number of modes
NP	NP			Number of lifting surface panels
NR	N			Total number of rows in the WT matrix
YP	yp_s	IN	ARG	y-coordinate of the lifting surface strip 3/4-chord point
Y1	Y1			y-coordinate of the inboard edge of panel
ZP	zp_s			z-coordinate of 3/4-chord point of the lifting surface strip
Z1	Z1			z-coordinate of the inboard edge of panel
NDW				I/O unit number containing the Δw matrix
NTP	NTP			Number of lifting surface boxes
NWT				I/O unit number on which the WT matrix is written
CBAR	$\bar{c}$			Length of the reference chord

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
NOUT				I/O unit number for printing output on.
NCOEF				Number of modal coefficients in the array A.
NBARAY		IN	ARG	Array containing the number of the last box in each panel.
NSARRY	NS			Array containing the number of strips in a panel.
NB				Number of bodies.
IPRINT				Print flag.

Calling Subroutines MAIN

Equations

$$w_{iq} = \sum_m \sum_n a_{q_{mn}}^{p} \left\{ \left(\frac{\bar{y}_p}{\bar{c}}\right)^m \left[n\left(\frac{xp}{\bar{c}}\right)^{n-1} + 2\,i\,k_r \left(\frac{xp}{\bar{c}}\right)^n \right] \right\}$$

$$\bar{y}_p = \sqrt{(yp_s - (N8)Y1)^2 + (zp_s - (N8)Z1)^2}$$

The array describing the modal coefficients is an integer at least 7 to 8 digits long

8 7 6 5 4 3 2 1

where

The digits -	87	-	the mode number
	654	-	the panel number
	3	-	m in the w_{iq} equation
	2	-	n in the w_{iq} equation
	1	-	N8 in the $\bar{y}_p$ equation

5.7 Segment 7

5.7.1 SUBROUTINE SOLVIT (A, RA, ND, MD, KD, NI, NM, NO, NW, NPR1)

Functional Description

Subroutine SOLVIT solves the system of simultaneous linear equations represented by the augmented rectangular (n x m)* matrix [DT|WT], which is written on logical tape unit NI, row by row, in the MAIN program. All other information necessary for the operation of SOLVIT, is entered via its argument list. The solutions obtained by SOLVIT are saved on logical tape unit NW in column order, i.e., one set of solutions per record; the data, entered from input tape NI is not preserved.

A detailed description of Subroutine SOLVIT can be found in Reference 3. Here we give only a brief description of the variables in the argument list.

Input Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
A(10000)				Complex work array
RA(2,10000)				Equivalenced real work array in which all the computations are done
ND	n			Total number of unknowns, i.e., size of the square DT-matrix
MD		IN	ARG	Total number of righthand sides in the system of simultaneous linear equations solved in SOLVIT
KD				Work array size (total real-variable dimension) – present value is 8000
NI				Tape number assigned to logical tape unit containing all rows of the augmented matrix [DT|WT]

*where n = number of unknowns and m = n + the number of right hand sides for which the solutions are obtained.

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
NM				Tape numbers, used as scratch units
NO		IN	ARG	
NW				Tape number assinged to tape containing all solutions
NPR1,N1				Print flag for solutions 0 - no print 1 - print

For the present assignment of tape units, see Section 4.1.

Calling Subroutine MAIN

5.8 Segment 8

5.8.1 SUBROUTINE BFSMAT (ND, NE, NB, NP, NTP, NTOTAL, IO, IPRNT, NAS, FMACH, YB, ZB, YS, ZS, X, DELX, EE, XIC, SG, CG, AR, RIA, NBEA1, NBEA2, NASB, NSARAY, NCARAY, BFS, AVR, CBAR, AO, XIS1, XIS2, KR, NSBEA, IBFS)

Functional Description

This subroutine is the basic calling routine that forms the $[FZ]^{(b)}$ and $[FY]^{(b)}$ matrices. (See equation (2.6-35 of Vol. 1. These matrices are written on tape a row at a time, alternating first a row of FZ then one of FY, etc. This set of matrices is formed for each body. The element FZ_{ij} gives the force in the z-direction due to either (1) a lifting surface box or (2) an axial doublet. The element FY_{ij} gives the force in the y-direction. The formulas differ depending on whether a lifting surface box or doublet is considered. The point pressure doublet is $\Delta C_p \delta A$ for the case of the lifting surface boxes. The point pressure doublet for an axial doublet involves a derivative with respect to x.

Input Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
ND	δ			Symmetry flag
NE	ε			Ground effect flag
NB				Number of bodies
NP				Number of panels
NTOTAL	$2\sum_{i=1}^{NB} NSBEA_i$	IN	ARG	2x (total number of slender body elements)
IO				Logical tape unit on which rows of the BFS matrix are written
IPRNT				Print flag
NAS				Array containing the number of associated bodies for each panel

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
FMACH				Mach Number
YB	y_B			Array of y-coordinates of the bodies
ZB	z_B			Array of z-coordinates
YS	y			Array of y-coordinates of strips and bodies
ZS	z			Array of z-coordinates
X	x			Array of the 3/4-chord locations of boxes and midpoints of interference body elements
DELX	Δx			Array of lengths of boxes and interference body elements
EE	e	IN	ARG	Array of the semi-widths of strips
XIC	ξ_c			Array of 1/4-chord locations of all boxes
SG	$\sin\gamma_s$			Array of the sines of the dihedral angles of strips
CG	$\cos\gamma_s$			Array of the cosines of the dihedral angles of strips
AR	$A\!\!R$			Array of the cross sectional aspect ratios of the bodies
RIA				Array of the radii of interference body elements
NBEA1				Array of the number of interference body elements per body
NBEA2				Z-y orientation flag array per body

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
NASB				Array of the bodies associated with panels
		IN	ARG	
NSARAY				Array of the number of strips per panel
NCARAY				Array of the number of chordwise boxes per panel
BFS				One row of the z- and y-forces in the BFS matrix
AVR				Array of the average radii of all bodies
CBAR	$\bar{c}$			Reference chord
AO	a_o			Array of the radii of slender body elements
XIS1	$\xi S1$			Array of the slender body element leading edge coordinates
XIS2	$\xi S2$			Array of the slender body element trailing edge coordinates
KR	k_r	IN	ARG	Reduced frequency
NSBEA				Array of the number of slender body elements per body
IBFS				Option flag = 1 to select subroutine FZY2 for the computation of the individual force element contributions

Calling Subroutine MAIN

Called Subroutine FWMW

Equations

The forces in the z- and y-directions on the slender body elements are related to the lifting surface box pressures, Δc_p, and axial doublet strengths $\tilde{\mu}^{(z)}, \tilde{\mu}^{(y)}$ through the [FZ] and [FY] matrices.

$$\left\{F_z\right\}^{(b)} = [FZ]^{(b)} \left\{p\right\}$$

$$\left\{F_y\right\}^{(b)} = [FY]^{(b)} \left\{p\right\}$$

where $\{p\} = \left\{\begin{array}{c} \Delta c_p \\ \overline{\tilde{\mu}^{(z)}} \\ \overline{\tilde{\mu}^{(y)}} \end{array}\right\}$

The elment $FZ_{ij} = FWMW^{(z)}_{ij} \Delta A_j$ when the sending element is a lifting surface box.

$$FZ_{ij} = FWMW^{(z)}_{ij} e^{-ik\frac{\Delta\xi}{\bar{c}} j} - FWMW^{(z)}_{ij+1} e^{+ik\frac{\Delta\xi}{\bar{c}} j+1}$$

where the superscript (z) on $FWMW_{ij}$ indicates that the force in the z-direction is desired, i.e., IF1 = 1. If the y-direction is desired set IF1 = 3. The subtraction indicated above numerically performs the differentiation required of $\tilde{\mu}^{(z)}$ and $\tilde{\mu}^{(y)}$.

5.8.2 SUBROUTINE FZY2 (XIJ, X1, X2, ETA, ZETA, YB, ZB, A, BETA2, CBAR, K, FZZR, FZZI, FZYR, FZYI, FZYR, FZYI, FYYR, FYYI, MFLG)

Functional Description

This subroutine calculates the forces in the z- and y-directions on all slender body elements of circular cross sections due to a unit pressure doublet located either inside or outside of the cross section. Subroutine FZY2 is called by subroutine FWMW, which in turn is called by BFSMAT and main only if the option flag IBFS = 1 (card input); it is

bypassed for all other cases (IBFS = 0).

Input Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
XIJ	ξ_j	IN	ARG	1/4-chord x-coordinate of slender body element
X1	x_1	IN	ARG	Leading edge of slender body element
X2	x_2	IN	ARG	Trailing edge of slender body element
ETA	η	IN	ARG	y-coordinate of sending point
ZETA	ζ	IN	ARG	z-coordinate of sending point
YB	y_B	IN	ARG	y coordinate of body centerline
ZB	z_B	IN	ARG	z coordinate of body centerline
A	a	IN	ARG	Radius of slender body element
BETA2	β^2	IN	ARG	$1-M^2$, where M = Mach Number
CBAR	$\bar{c}$	IN	ARG	Reference chord
K, KR	k_r	IN	ARG	Reduced frequency
FZZR		OUT	ARG	Real part of the $F_z^{(z)}$ Eq. (5.8.2-1)
FZZI		OUT	ARG	Imaginary part of the $F_z^{(z)}$ Eq. (5.8.2-1)
FZYR		OUT	ARG	Real part of the $F_z^{(y)}$ Eq. (5.8.2-2)
FZYI		OUT	ARG	Imaginary part of the $F_z^{(y)}$ Eq. (5.8.2-2)

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
FYYR		OUT	ARG	Real part of the $F_y^{(y)}$ Eq. (5.8.2-3)
FYYI				Imaginary part of the $F_y^{(y)}$ Eq. (5.8.2-3)
MFLG		IN	ARG	Detail print flag
TEST1		IN	FZY2	The number 1/7
TEST 2				The number 1/2
KBAR	$\bar{k}$			Eq. (5.8.2-9)
DX	Δx			$x_2 - x_1$
RAA	R_{aA}			Eq. (5.8.2-6)
DELTA	δ			$\Delta x/R_{aA}$
XA	x_a			$(x_1 + x_2)/2$
RWIG	$\tilde{r}$			Eq. (5.8.2-7)
RAIJ	ra_{ij}			Eq. (5.8.2-5)
QR	Re(Q)			Eq. (5.8.2-8)
QI	Im(Q)			
CAPA	A			Eq. 5.8.2-15)
CAPDR	Re(Δ)			Eq. (5.8.2-12) or Eq. (5.8.2-16)
CAPDI	Im(Δ)			
FTHR	$Re(f_\theta^{(\theta)})$			Eq. (5.8.2-10) or Eq. (5.8.2-13)
FTHI	$Im(f_\theta^{(\theta)})$			
FRR	$Re(f_r^{(r)})$			Eq. (5.8.2-11) or Eq. (5.8.2-14)
FRI	$I_m(f_r^{(r)})$			

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
CTH STH	$\cos\theta$ $\sin\theta$	IN	FZY2	Eq. (5.8.2-4)
I1 through I11	I_1 through I_{11}			The integrals I_1 through I_{11}

Calling Subroutine FWMW

Equations

Subroutine FZY2 computes the unsteady body forces for circular cross sections according to the following equations.

$$FZZ_{ij} = \cos^2\theta f_\theta^{(\theta)} + \sin^2\theta f_r^{(r)} \qquad (5.8.2\text{-}1)$$

$$FZY_{ij} = FYZ_{ij} = \cos\theta \sin\theta \, (f_r^{(r)} - f_\theta^{(\theta)}) \qquad (5.8.2\text{-}2)$$

$$FYY_{ij} = \sin^2\theta f_\theta^{(\theta)} + \cos^2\theta \, f_r^{(r)} \qquad (5.8.2\text{-}3)$$

where

$$\cos\theta = \frac{\eta_j - YB_i}{r_{aij}} \, , \quad \sin\theta = \frac{\zeta_j - ZB_i}{r_{aij}} \qquad (5.8.2\text{-}4)$$

and

$$r_{aij}^2 = (\eta_j - YB_i)^2 + (\zeta_j - ZB_i)^2 \qquad (5.8.2\text{-}5)$$

where i is the index of the receiving body element
and j is the index of the sending element.
Note that we set
$\cos\theta = 1.0$ and $\sin\theta = 0.0$ whenever $r_{aij} = 0.0$.

The formulae for $f_\theta^{(\theta)}$ and $f_r^{(r)}$ are calculated depending on the location of the sending element, as shown below.

Define a number δ as follows

$$\delta = \Delta x / R_{aA}$$

where

$$R_{aA} = \sqrt{(x_A - \xi_j)^2 + \beta^2 \tilde{r}^2} \qquad (5.8.2\text{-}6)$$

$$\tilde{r}^2 = \begin{cases} r_a{}^2{}_{ij} & \text{if } r^2_{aij} > a^2 \\ a^2 & \text{otherwise} \end{cases} \qquad (5.8.2\text{-}7)$$

$$\Delta x = x_{2j} - x_{1i}$$

$$x_A = \frac{x_{1i} + x_{2i}}{2}$$

Also, define

$$Q = \frac{1}{4\Delta x} e^{\, i \frac{\bar{k}}{\beta^2 a} [M(x_A - \xi_j) - R_{aA}]} \qquad (5.8.2\text{-}8)$$

where

$$\bar{k} = 2\ k\ M\ a_i / \bar{c} \qquad (5.8.2\text{-}9)$$

Then, if $\delta \leq 1/7$

$$f_\theta^{(\theta)} = Qa(\beta^2 a I_1 + i\bar{k}\ I_4) \qquad (5.8.2\text{-}10)$$

and

$$f_r^{(r)} = f_\theta^{(\theta)} + \Delta \qquad (5.8.2\text{-}11)$$

where

$$\Delta = \begin{cases} Qr_{aij}^2[-3\beta^2 a(\beta^2 aI_6 + i\bar{k}\ I_9) + \bar{k}^2 I_1] & \text{if } r_{aij}^2 > a^2 \\ 0 & \text{otherwise} \end{cases} \tag{5.8.2-12}$$

and

$$I_1 = \delta/R_{aA}^2$$

$$I_4 = \delta/R_{aA}$$

$$I_6 = \delta/R_{aA}^4$$

$$I_9 = \delta/R_{aA}^3$$

If $\delta > 1/7$

$$f_\theta^{(\theta)} = Qa[(\beta^2 aI_1 - \frac{\bar{k}^2}{\beta^2 a} AI_5) + i\bar{k}(AI_2 + I_4 - \frac{I_3\beta^2\tilde{r}^2}{2R_a^3 A} \tag{5.8.2-13}$$

$$f_r^{(r)} = f_\theta^{(\theta)} + \Delta \tag{5.8.2-14}$$

where

$$A = M-(x_A - \xi_j)/R_{aA} \tag{5.8.2-15}$$

and

$$\Delta = \begin{cases} Qr_{aij}^2 \quad -3\beta^4 a^2 I_6 + \bar{k}^2(I_1 + 3AI_{1o}) & \\ + i3\beta^2 a\bar{k}(-AI_7 + \dfrac{I_8\beta^2\tilde{r}^2}{2R_a^3 A} - I_o) + i\dfrac{k^{-3}\ AI_2}{\beta^2 a} & \text{if } r_{aij}^2 > a^2 \\ 0 \quad \text{otherwise} & \end{cases} \tag{5.8.2-16}$$

and where the integrals I_1 through I_{10} are calculated two different ways, depending on the value of the number δ:

If $\quad \delta \leq 1/2$

$$I_1 = \frac{\delta}{R_{aA}^2} \quad 1 - \left[\frac{\delta^2}{8}(-1 + 5\tau^2)\right]$$

$$I_2 = \frac{\delta^3}{R_{aA}} (-\tau/4)$$

$$I_3 = \delta^3/12$$

$$I_4 = \frac{\delta}{R_{aA}} \left[1 + \frac{\delta^2}{12}(-1 + 3\tau^2)\right]$$

$$I_5 = -\tau\delta^3/6$$

$$I_6 = \frac{\delta}{R_{aA}^4} \left[1 + \frac{5\delta^2}{24}(-1 + 7\tau^2)\right]$$

$$I_7 = \frac{\delta^3}{R_{aA}^3} \left(-5\tau/12\right)$$

$$I_8 = \delta^3/(12\ R_{aA}^2)$$

$$I_9 = \frac{\delta}{R_{aA}^3} \left[1 + \frac{\delta^2}{6}(-1 + 6\tau^2)\right]$$

$$I_{10} = \frac{\delta^3}{R_{aA}^2} (-\tau/3)$$

where

$$\tau = \frac{x_A - \xi_j}{R_{aA}}$$

while, if $\delta > 1/2$

$$I_1 = \frac{1}{\beta^2 \tilde{r}^2} \left[\frac{(x_2 - \xi)}{R_{a2}} - \frac{(x_1 - \xi)}{R_{a1}}\right]$$

$$I_2 = -\frac{1}{\beta^2 \tilde{r}^2} \left[\frac{(x_A - \xi)\Delta x_2 + R_{aA}^2}{R_{a2}} + \frac{(x_A - \xi)\Delta x/2 - R_{aA}^2}{R_{a1}}\right]$$

$$I_{11} = \ln\left|\frac{x_2 - \xi + R_{a2}}{x_1 - \xi + R_{a1}}\right|$$

$$I_3 = I_{11} - 2(x_A - \xi)I_2 - R_{aA}^2\, I_1$$

$$I_4 = \frac{1}{\beta\tilde{r}}\left[\text{Atan}\ \frac{x_2 - \xi}{\beta\tilde{r}} - \text{Atan}\ \frac{x_1 - \xi}{\beta\tilde{r}}\right]$$

$$I_5 = \frac{1}{2}\ln\left(\frac{R_{a2}^2}{R_{a1}^2}\right) - (x_a - \xi)I_4$$

$$I_6 = \frac{1}{3\beta^2\tilde{r}^2}\left[\frac{(x_2 - \xi)}{R_{a2}^3} - \frac{(x_1 - \xi)}{R_{a1}^3} + 2\ I_1\right]$$

$$I_7 = -\frac{1}{3}\left(\frac{1}{R_{a2}^3} - \frac{1}{R_{a1}^3}\right) - (x_A - \xi)I_6$$

$$I_8 = I_1 - 2(x_A - \xi)I_7 - R_{aA}^2\, I_6$$

$$I_9 = \frac{1}{2\beta^2\tilde{r}^2}\left[\frac{(x_2 - \xi)}{R_{a2}^2} - \frac{(x_1 - \xi)}{R_{a1}^2} + I_4\right]$$

$$I_{10} = -\frac{1}{2\beta^2\tilde{r}^2}\left[\frac{(x_A - \xi)\Delta x/2 + R_{aA}^2}{R_{a2}^2} + \frac{(x_A - \xi)\Delta x/2 - R_{aA}^2}{R_{a1}^2} + (x_A - \xi)I_4\right]$$

where

$$R_{a1} = \sqrt{(x_1 - \xi)^2 + \beta^2\tilde{r}^2} \quad \text{and} \quad R_{a2} = \sqrt{(x_2 - \xi)^2 + \beta^2\tilde{r}^2}$$

5.9 Segment 9

5.9.1 SUBROUTINE BFM (IWORK, RWORK, WORK, NWORK, NPTAP, NPSTAP, NBFM, NPOT, NM, IPRNT, IERRØR, IBFS)

Functional Description

Each and every singularity in the flow field, whether it be inside or outside of a body, contributes to the force distribution on a body.

Subroutine BFM determines this loading, force and moment, on bodies that occurs due to these singularities. The following are the flow singularities considered.

(a) Slender body singularities.
(b) Interference element singularities.
(c) Lifting surface boxes external to the body.
(d) Lifting surface box images both inside and outside of the body.
(e) The additional contributions due to considerations of symmetry and ground effects for the above singularities.

Subroutine BFM is essentially a calling routine. Generally speaking, FWMW determines the loading on the body for a pressure singularity of unit strength, including images, symmetry, and ground effect. Subroutine ORGAN gives the region, on the body, over which the loading acts. Subroutine SBLOAD determines the pressure singularity strength from the various flow singularity strengths and gives the detailed loads on each slender body element. BFM then uses the unit loading obtained from FWMW and the results from SBLOAD in conjunction with ORGAN to find the load (both forces and moments) on each of the slender body elements. The slender body elements have been chosen as a convenient set of elements over which all body forces are distributed.

Input Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
AR	$\mathcal{R}$			Aspect ratio of the bodies.
CG	$\cos(\gamma)$			Cosine dihedral angle.
EE	e	IN	Blank Common Block	
NB	NB			Number of bodies.
ND	δ			Symmetry flag
NE	ε			Ground effects flag
NM	NM	IN	ARG	Number of modes.
NP	NP			Number of panels.
SG	$\sin(\gamma)$			Sine dihedral angle.
YB	y_c			y coordinate of body center.
YP	y			y coordinate of centerline of strip
ZB	z_c	IN		z coordinate of body center.
ZP	z			z coordinate of centerline of strip
AVR				Average radius of bodies
NAS				Number of associated bodies per panel
NBY				Number of y-oriented bodies.
NBZ				Number of z-oriented bodies.
NTO		IN	Blank Common Block	Total number of boxes and interference body elements.
NTP				Total number of boxes
NTY				Number of y-oriented interference body elements
NTZ				Number of z-oriented reference body elements
XLE	x_{LE}	IN		Leading edge of bodies
XTE	x_{LE}			Trailing edge of bodies
DELX				Length of boxes and interference body elements
MACH	M			Mach number
NASB				Number of the associated bodies per panel
NBFA				Number of interference body elements per body and body orientation.
IPRNT				Print flag
IERRØR		IN	ARG	Error flag
IBFS				Body force calculation method flag

MNEMONICS	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
NBFM		IN	ARG	I/O unit which output body and forces are to be written on (unformatted)
NPOT				Print output unit
NSBE				Number of slender body elements per body
NTYS	NTYS			Number of y-oriented slender body elements
NTZS	NTZS			Number of z-oriented slender body elements
WORK				Complex array for temporary storage of results. The length in complex words required is 4 (No. Strips + NB2 + NBY) + 4(NTZS + NTYS)
XIS1	ξ_{S_1}			Leading edge of slender body elements
XIS2	ξ_{S_2}	IN		Training edge of slender body elements
IWORK			ARG	Integer array for temporary storage of results its length is 2(NTP + NTZ + NTY + NTZS + NTYS) words
NPTAP				I/O unit containing the P matrix
NWORK				Size of the array WORK
RWORK				Real array for temporary storage. The length is = NTP + 1 + 2 {maximum [(NTP + NTZ + NTY) or (NTZS + NTYS)]}
NCARRY				Array containing the number of chordwise boxes per strip in a panel
NPSTAP		IN	ARG	I/O unit number containing the PS matrix
NSARRY				Array containing the number of strips in a panel.

Calling Subroutines MAIN

Called Subroutines and Common Blocks

RWREC, WRTFMU, FWMW, ORGAN, SBLOAD, WRTFMF, Blank Common

Error Messages

SUBROUTINE **BFM** NWORD(NNNNNNNN) IS GREATER THAN NWORK(NNNNNNNN)

JOB TERMINATED

The available working array is too small to execute the subroutine. Either decrease the problem requirements or increase the size of the work area.

5.9.2 SUBROUTINE FMZY (DYB, DZB, DA, DAR, DY, DZ, DKR, DM, DCBAR, DRFZZ, DIFZZ, DRFZY, DIFZY, DRFYZ, DIFYZ, DRFYY, DIFYY, DRMZZ, DIMZZ, DRMZY, DIMZY, DRMYZ, DIMYZ, DRMYY, DIMYY, IF1, IF2)

Functional Description

This subroutine calculates the force and moment on an elliptic cross section due to a unit pressure doublet located either inside or outside of the cross section.

Input Output Variables

MNEMONICS	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
DYB DZB	YB ZB	IN	ARG	Origin of ellipse
DA	a			Width of ellipse in y-direction
DAR	AR=b/a			Ratio of semi height to semi width of the ellipse
DY DZ	y and $\bar{y}$ z and $\bar{z}$			Coordinates of pressure doublet. For bared quantity $\bar{y}$=y-yB, $\bar{z}$=z-zB
DKR	k			Reduced frequency, $\omega\bar{c}/2U_\infty$
DM	M			Mach number
DCBAR	$\bar{c}$			Reference chord length
DRFZZ DIFZZ	$F_z^{(z)}$	OUT	ARG	Real and imaginary parts of z-force due to doublet oriented in z-direction
DRFZY DIFZY	$F_z^{(y)}$			Real and imaginary parts of z-force due to y-doublet
DRFYZ DIFYZ	$F_y^{(z)}$			Real and imaginary parts of y-force due to z-doublet
DRFYY DIFYY	$F_y^{(y)}$			Real and imaginary parts of y-force due to y-doublet
DRMZZ DIMZZ	$M_y^{(z)}$			Real and imaginary parts of z-moment due to z-doublet
DRMZY DIMZY	$M_z^{(y)}$			etc
DRMYZ DIMYZ	$M_y^{(z)}$			etc
DRMYY DIMYY	$M_y^{(y)}$			etc

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
IF1		IN	ARG	1 if body z-oriented, 2 if body z and y oriented and 3 if y-oriented
IF2		IN	ARG	0 if doublet outside of ellipse 1 if doublet inside of ellipse
DTHETO	Θ_o			Elliptic coordinate at which integration starts
FMUR FMUI	f			Real and imaginary parts of f
PMUR PMUI	p			Real and imaginary parts of p
DKJO DKYO DKJ1 DKY1	$kJ_o(\bar{k})$ $kY_o(\bar{k})$ $kJ_1(\bar{k})$ $kY_1(\bar{k})$	IN	FMZY	J_γ, Y_γ are Bessel function of order γ
EF1 EF2	$\cos\lambda$ $\sin\lambda$			λ is the surface slope of the elliptic cross section at the point
DKK	$\bar{k}$			
DRR	R			Distance from field point to surface point
RCURV	$\bar{a}$			Surface curvature of the ellipse
DEE	e			Element half width

Calling Subroutines FWMW

Equations

If $k\dot{M} = 0$ and $AR = 1$ and $r/a > 1$

$$F_z^{(z)} = (\bar{y}^2 - \bar{z}^2)\,\frac{1}{2}\,\frac{a^2}{r^4}$$

$$F_z^{(y)} = -(\bar{z}\,\bar{y})\,a^2/r^4$$

$$F_y^{(z)} = F_z^{(y)}$$

$$F_y^{(y)} = F_z^{(z)}$$

$$M_z^{(z)} = M_z^{(y)} = M_y^{(z)} = M_y^{(y)} = 0$$

$$\bar{z} = z - zB$$

$$\bar{y} = y - yB$$

If $kM = 0$ and $r/a < 1$

$$F_z^{(z)} = 1/(1 + AR)$$

$$F_y^{(y)} = AR/(1 + AR)$$

$$F_z^{(y)} = F_y^{(z)} = M_z^{(z)} = M_z^{(y)} = M_y^{(z)} = M_y^{(y)} = 0$$

Otherwise numerical integration is performed

$$\Theta_o = \Theta_1 \quad \text{if } \bar{z} \text{ and } \bar{y} > 0$$

$$\Theta_o = -\Theta_1 \quad \text{if } \bar{y} > 0 \text{ and } z < 0$$

$$\Theta_o = \Theta_1 + \pi \text{ if } \bar{y} < 0 \text{ and } \bar{z} < 0$$

$$\Theta_o = \pi - \Theta_1 \quad \text{if } \bar{y} < 0 \text{ and } \bar{z} > 0$$

$$\cos^2\Theta_1 = 1 - \sin^2\Theta_1$$

$$\sin^2\Theta_1 = \frac{-A + \sqrt{A^2 + 4a^2(1 - AR^2)(z - zB)^2}}{2a^2(1 - AR^2)}$$

$$A = (y - yB)^2 + (z - zB)^2 + a^2(AR^2 - 1)$$

If $|1 - AR^2| < .001$

$$\sin^2\Theta_1 = (z - zB)^2 / r^2$$

$$\Theta = \Theta_1 + \Delta\Theta$$

$$\Delta\Theta = 2\pi / N \qquad \text{(Currently N = (NUMB) = 30)}$$

$$F_z^{(z)} = \sum_{j=1}^{N} f(\bar{k}_j)\, EFZZ_j$$

$$F_z^{(y)} = \sum_{j=1}^{N} f(\bar{k}_j)\, EFZY_j$$

$$F_y^{(z)} = \sum_{j=1}^{N} f(\bar{k}_j)\, EFYZ_j$$

$$F_y^{(y)} = \sum_{j=1}^{N} f(\bar{k}_j)\, EFYY_j$$

$$M_z^{(z)} = \sum_{j=1}^{N} P(\bar{k}_j)\, EF1_j EIM2_j$$

$$M_z^{(y)} = \sum_{j=1}^{N} P(\bar{k}_j)\, EF1_j EIM1_j$$

$$M_y^{(z)} = \sum_{j=1}^{N} - P(\bar{k}_j)\, EF2_j\, EIM2_j$$

$$M_y^{(y)} = \sum_{j=1}^{N} - P(\bar{k}_j)\, EF2_j\, EIM1_j$$

$$f(\bar{k}_j) = -\frac{1}{4}\bar{k}_j \left\{ Y_1(\bar{k}_j) + i\ J_1(\bar{k}_j) \right\}$$

$$P(\bar{k}_j) = \frac{M}{4}\bar{k}_j \left\{ J_o(\bar{k}_j) - i\ Y_o(\bar{k}_j) \right\}$$

Y_ν and J_ν are Bessel functions of order ν and are approximated by polynomials taken from Reference 4

$$\bar{k} = 2\ k\ M\ R/\bar{c}$$

$$EF1 = \cos\lambda$$

$$EF2 = \sin\lambda$$

$$EIM1 = 2e\ (y - \eta)/R$$

$$EIM2 = 2e\ (z - \zeta)/R$$

$$R^2 = (\bar{y} - \eta)^2 + (\bar{z} - \zeta)^2$$

$$e = -\frac{\pi}{N}\ \frac{1}{AR} \sqrt{\zeta^2 + AR^4\ \eta^2}$$

$$\eta = a \cos\Theta$$

$$\zeta = b \sin\Theta$$

$$\cos\lambda = \frac{\zeta}{\sqrt{\zeta^2 + AR^4\ \eta^2}}$$

$$\sin\lambda = \frac{-\ AR^2\ \eta}{\sqrt{\zeta^2 + AR^4\ \eta^2}}$$

If $R/e \geq 5.0$

$$EF = \bar{k}\ \frac{2e}{R^2}$$

IF $1.5 \geq R/e < 5.0$

$$EF = \bar{K}\ \frac{2e}{R^2} + \frac{2}{3}\left(\frac{e}{R}\right)^3 \left[I\ \frac{R}{\bar{a}} - \frac{B\bar{J}}{R} + \frac{\bar{K}}{R}\ \left(\frac{D}{\bar{\bar{a}}} - 1 + \frac{B^2}{R^2}\right)\right]$$

$$+ \frac{2}{5} \left(\frac{e}{R}\right)^5 \left[\frac{\bar{K}}{R} \left\{ - \left(\frac{R}{\bar{a}}\right)^2 \frac{1}{4} + 1 \frac{-2D}{\bar{a}} + \frac{D^2}{\bar{a}^2} - 3 \left(\frac{B}{R}\right)^2 (1 - D/\bar{a}) + \left(\frac{B}{R}\right)^4 \right\}\right.$$

$$\left. + \bar{J} \left\{ 2 \frac{B}{R} (1 - D/\bar{a}) - (B/R)^3 \right\} + I \frac{R}{\bar{a}} \left\{ (B/R)^2 - 1 + D/\bar{a} \right\} \right]$$

IF $R/e < 1.5$

$$EF = \left\{ - \frac{R^2 I}{\bar{a}} - \frac{\bar{J}B}{2} + \bar{K} - \frac{3}{4} \frac{D\bar{J}B}{\bar{a}} + \frac{D\bar{K}}{2\bar{a}} \right\} \frac{1}{R} \text{ Atan } \left(\frac{2Re}{R^2 - e^2} \right)$$

$$+ \frac{2eI}{\bar{a}} + \frac{1}{e^2 + R^2} \left\{ eB\bar{J} (1 + D/\bar{a}) + \frac{D\bar{J}Be(3e^2 + R^2)}{2\bar{a}(e^2 + R^2)} \frac{-D\bar{K}e}{\bar{a}} \right\}$$

$$EFZZ = EF \text{ with } \begin{cases} I = 3/2 \sin^2\lambda - 1/2 \\ \bar{J} = \sin\lambda\cos\lambda - \text{in}\lambda \left(\frac{\bar{z}-\zeta}{\bar{a}}\right) \\ \bar{K} = -\cos\lambda\,(\bar{z} - \zeta) \end{cases}$$

$$EFYZ = EF \text{ with } \begin{cases} I = 3/2 \cos\gamma\sin\gamma \\ \bar{J} = \cos^2\lambda - \sin\lambda \frac{(\bar{y} - \eta)}{\bar{a}} \\ \bar{K} = -\cos\lambda\,(\bar{y} - \eta) \end{cases}$$

$$EFYZ = EF \text{ with } \begin{cases} I = 3/2 \cos\lambda\sin\lambda \\ \bar{J} = -\sin^2\lambda - \cos\lambda \frac{(\bar{z} - \zeta)}{\bar{a}} \\ \bar{K} = \sin\lambda\,(z - \zeta) \end{cases}$$

$$EFYY = EF \text{ with } \begin{cases} I = 3/2 \cos^2\lambda - 1/2 \\ \bar{J} = -\cos\lambda\sin\lambda - \cos\lambda \frac{(\bar{y} - \eta)}{\bar{a}} \\ \bar{K} = \sin\lambda\,(\bar{y} - \eta) \end{cases}$$

$$B = -2\,(\cos\lambda\,(\bar{y} - \eta) + \sin\lambda\,(\bar{z} - \zeta))$$

$$D = \sin\lambda\,(\bar{y} - \eta) - \cos\lambda\,(\bar{z} - \zeta)$$

$$\bar{a} = \frac{a}{AR} \left\{ \sin^2\Theta + AR^2\cos^2\Theta \right\}^{2/3}$$

5.9.4 SUBROUTINE ORGAN (ISTART, ISTOP, NLBE, NTP, X, DELX, PERCNT, XLE, XTE, XIS1, XIS2, ITYPE)

Functional Description

It is assumed that the effect of a flow singularity in a body acts exactly at the same x-location as the flow singularity itself. This routine therefore generates two output arrays, one identifying the slender body elements which the leading edges of the sending elements lie within and the other the trailing edges.

Input Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
X	x	IN	ARG	ITYPE≠1 Array of sending box or body element location ITYPE = 1 Array of sending slender body elements leading edge x-location.
NTP	NTP	IN	ARG	Number of boxes or elements sending loads
XLE	$x_{L.E.}$			Leading edge of receiving slender body
XTE	$x_{T.E.}$			Trailing edge of receiving slender body
DELX		IN	ARG	ITYPE≠1 Array of sending box or body element length ITYPE = 1 Array of sending slender body elements trailing edge x-location
NLBE		IN	ARG	Number of elements in the receiving slender body.
XIS1	ξ_{s_1}			Array of leading edge x-coordinates of the receiving slender body elements
XIS2	ξ_{s_2}			Array of the x-coordinates of the trailing edges of the elements of the receiving slender body

ITYPE		IN	ARG	Flag indicating the type of the sending elements <u>≠ 1</u> sending elements are either boxes or interference bodies. <u>= 1</u> sending elements are slender bodies.
ISTART		OUT	ARG	An array containing the first slender body element receiving a load contribution from the sending element. The length of this array is (NTP)
ISTOP		OUT	ARG	An array containing the last slender body element receiving a load contribution from the sending element. The length of this array is (NTP).
PERCNT		IN	ARG	The fractional location of the input x with respect to the length of the sending element. This is used to calculate the leading edge of the sending element.

<u>Calling Subroutine</u> BFM

5.9.5 <u>SUBROUTINE SBLOAD (COEF, IFIRST, ILAST, KS, XIS1, XIS2, DELTA, N, PERCNT, XLE, XTE, NBE, FWZ, FWY, MWZ, MWY, FZ, FY, MZ, MY</u>

<u>Functional Description</u>

It is assumed that the x-distribution of the flow singularity is constant over the flow of the flow singularity element. It is further assumed that the effect of a flow singularity on a bcdy acts exactly at the same x-location as the flow singularity itself. This routine distributes the forces and moments on a body due to internal and external flow singularities. The center of force in each of the slender body elements is also calculated. The output from subroutine ORGAN determines which slender body elements are affected.

Input Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
X		IN	ARG	x-coordinate of sending element or box
FY	F_y	OUT	ARG	y-force array
FZ	F_z	OUT	ARG	z-force array
KS		IN	ARG	reserved
MY	M_y	OUT	AGR	y-moment (yawing) array
MZ	M_z	OUT	ARG	z-moment (pitching) array
FWY	F_{wy}			unit y-force on sending body or strip
FWZ	F_{wz}	IN	ARG	unit z-force on sending body or strip
MWY	M_{wy}			unit y-moment on sending body or strip
MWZ	M_{wz}			unit z-moment on sending body or strip
NBE	N			Number of body elements or receiving body.
XLE		IN	ARG	Reserved
XTE				Reserved
COEF				
XIS1	ξ_{s1}			Array of leading edge x-coordinate
XIS2	ξ_{s2}	IN	ARG	Array of trailing edge x-coordinates of the slender body elements.
DELTA	Δx			Length of sending box or element.
ILAST				Last slender body element to which the loads are to be applied.
PERCNT				PERCNT = $(x - x_{LE})$/DELTA where x_{LE} is the x-coordinate of the leading edge of the sending element or box.

Calling Subroutines BFM

5.10 Segment 10

5.10.1 SUBROUTINE AERO (NMODE, NSTRIP, NW, NBFM, NEWBFM, IBFS)

Functional Description

Subroutine AERO computes the aerodynamic parameters for each mode. These include the sectional lift and moment coefficients for all strips of all lifting surface panels, and the center of pressure locations (Eqs. 5.10.1-1 through 5.10.1-3); lift and moment coefficients (z- and y-components) for all slender body elements (Eqs. 5.10.1-4 through 5.10.1-9); total lift and moment coefficients for each slender body (Eqs. 5.10.1-10 through 5.10.1-13); and the total lift and moment coefficients including body effect (Eqs. 5.10.1-14 through 5.10.1-18). The above aerodynamic parameters are printed along with the strip number, or body element number in case of a slender body, and the y- and z-coordinates of the strips and slender bodies.

Input Output Variables

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
NMODE				Total number of modes
NSTRIP				Total number of strips on all lifting surface panels
NW		IN	Argument List of AERO	Tape number for logical tape unit containing the solutions (ΔC_p) for all unknowns* 'n' and all modes
NBFM				Tape number for logical tape unit containing all slender body element forces and moments for all modes when IBFS = 0
NEWBFM				Tape number for logical tape unit containing all slender body element forces when IBFS = 1
IBFS				Body force calculations method flag
FZ(200)	$\partial f_z/q$			z-forces for all slender body elements and all modes
FY(200)	$\partial f_y/q$		Tape NBFM or NEWBFM	y-forces for all slender body elements and all modes
MZ(200)				z-moments for all slender body elements and all modes
MY(200)		IN		y-moments for all slender body elements and all modes

*n = total number of boxes + total number of interference body elements.

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
DCP(500)	ΔCp		Tape NW	Pressure coefficients (solutions) for all unknowns and all modes
CN(200)	$c_{n_{qj}}$			See Equation (5.10.1-1)
CM(200)	$c_{m_{qj}}$			See Equation (5.10.1-2)
CPR(200)	CPR_{qj}			See Equation (5.10.1-3a)
CPI(200)	CPI_{qj}			See Equation (5.10.1-3b)
FZLB	$c_z^{(b)'}$			See Equation (5.10.1-4)
FYLB	$c_y^{(b)'}$			See Equation (5.10.1-5)
MZLB	$c_{mz}^{(b)'}$			See Equation (5.10.1-6)
MYLB	$c_{my}^{(b)'}$	OUT	AERO	See Equation (5.10.1-7)
DFZX	$\frac{\partial f_{zt}/\partial x}{q}$			See Equation (5.10.1-8)
DFYX	$\frac{\partial f_{yt}/\partial x}{q}$			See Equation (5.10.1-9)
CZB(10)	$C_{Zq}^{(b)}$			See Equation (5.10.1-10)
CYB(10)	$C_{Yq}^{(b)}$			See Equation (5.10.1-11)
CMB(10)	$C_M^{(b)}$			See Equation (5.10.1-12)
CNB(10)	$C_N^{(b)}$			See Equation (5.10.1-13)
CMULT				$2e \cos \gamma$
SMULT				$2e \sin \gamma$
GUCJ	G_j			1 or 0.5; Equation (5.10.1-19)
GLCB	$g^{(b)}$	IN		1 or 0.5; Equation (5.10.1-20)
SYMA				$1 + \delta$ | where δ is the symmetry
SYMB			AERO	$1 - \delta$ | flag (input)
CZT	C_{Zq}			See Equation (5.10.1-14)
CYT	C_{Yq}	OUT		See Equation (5.10.1-15)
CMT	C_M			See Equation (5.10.1-16)
CNT	C_N			See Equation (5.10.1-17)
CLT	C_ℓ			See Equation (5.10.1-18)

Calling Subroutine MAIN

Common Blocks The Blank Common Block

Equations

The following aerodynamic parameters are calculated for each mode:

A. Lifting Surface Strips

$$c_{n_{qj}} = \frac{1}{c_j} \sum_{i=i_1}^{i_2} \Delta C_{pqi} \, \Delta x_i \qquad (5.10.1\text{-}1)$$

$$c_{m_{qj}} = \frac{-1}{c_j^2} \sum_{i=i_1}^{i_2} \Delta C_{pqi} \, \Delta x_i (\xi_{ci} - \xi 14_j) \qquad (5.10.1\text{-}2)$$

$$CPR_{qj} = \frac{Re(C_{m_{qj}})}{Re(C_{n_{qj}})} + \frac{1}{4} \qquad (5.10.1\text{-}3a)$$

$$CPI_{qj} = \frac{Im(C_{m_{qj}})}{Im(C_{n_{qj}})} + \frac{1}{4} \qquad (5.10.1\text{-}3b)$$

where

- i_1 and i_2 are the indices of the first and last boxes in strip j, where $j = 1$, NSTRIP;
- c_j is the chordlength of strip j, and
- $\xi 14_j$ is the x-coordinate of the 1/4-chord point on the centerline of strip j.

B. Slender Body Elements

$$c_z^{(b)}{}_t = \frac{f_{z_{qt}}^{(b)}}{\Delta xSB_t^{(b)}} \qquad (5.10.1\text{-}4)$$

$$c_y^{(b)\prime} = \frac{f_{y_{qt}}^{(b)}}{\Delta xSB_t^{(b)}} \qquad (5.10.1\text{-}5)$$

$$c_{m_z}^{(b)\prime} = \frac{m_{z_{qt}}^{(b)}}{\Delta xSB_t^{(b)}} \qquad (5.10.1\text{-}6)$$

and

$$c_{m_y}^{(b)\prime} = \frac{m_{y_{qt}}^{(b)}}{\Delta xSB_t^{(b)}} \qquad (5.10.1\text{-}7)$$

where

$\Delta xSB_t^{(b)} = \xi S2_t^{(b)} - \xi S1_t^{(b)}$ = length of slender body element 't' in body 'b'; see Blank Common, Sec. 3.1.

In addition to the above aerodynamic parameters, subroutine AERO also computes modified lift coefficients (z- and y-components) for all slender body elements according to the following equations.

$$\left(\frac{\partial f_{zt}/\partial x}{q}\right) = AR\left[\overline{GZ}_{t\overline{t}}\right]\left(\frac{\partial f_{z\overline{t}}}{q}\right) \qquad (5.10.1\text{-}8)$$

and

$$\left(\frac{\partial f_{yt}/\partial x}{q}\right) = \left[\overline{GY}_{t\overline{t}}\right]\left(\frac{\partial f_{y\overline{t}}}{q}\right) \qquad (5.10.1\text{-}9)$$

where the elements of the $NSBE^{(b)} \times NSBE^{(b)}$ matrices $\overline{GZ}_{t\overline{t}}$ and $\overline{GY}_{t\overline{t}}$ are defined as follows:

$$\overline{GZ}_{t\overline{t}} = \overline{G}(a^{(b)}, k, M, \overline{c}, AR, R = RZ_{t\overline{t}}, x - \xi = xSB_t - xSB_{\overline{t}})$$

$$\overline{GY}_{t\overline{t}} = \overline{G}(a^{(b)}, k, M, \overline{c}, AR, R = RY_{t\overline{t}}, x - \xi = xSB_t - xSB_{\overline{t}})$$

where

$$\overline{G}(a,k,M,\overline{c},AR,R,x-\xi) = \frac{a^2\beta^2}{R^2}\left(\frac{1 + AR}{4}\right)\left(\frac{1}{R} + i\,\frac{2kM}{\beta^2\overline{c}}\right)e^{i(2kMf/\beta^2)}$$

$$f = [(x - \xi)M - R]/\overline{c}$$

$$RZ_{t\bar{t}} = \sqrt{(x_t - \xi_{\bar{t}})^2 + \beta^2 a^2 AR(1 + AR)/2}$$

and

$$RY_{t\bar{t}} = \sqrt{(x_t - \xi_{\bar{t}})^2 + \beta^2 a^2 (1 + AR)/2}$$

The indices 't' and '$\bar{t}$' denote the 'row' and 'column' subscripts for the matrices $[\overline{GZ}_{t\bar{t}}]$ and $[\overline{GY}_{t\bar{t}}]$ corresponding to the slender body elements 't' and '$\bar{t}$', and $NSBE^{(b)}$ = number of slender body elements in body 'b'. The total lift and moment coefficients (z- and y-directions) for all slender bodies are defined by the following equations.

$$C_{Z_q}^{(b)} = \frac{1}{A} \sum_{t=1}^{NSBE^{(b)}} f_{z_{qt}}^{(b)} \tag{5.10.1-10}$$

$$C_{Y_q}^{(b)} = \frac{1}{A} \sum_{t=1}^{NSBE^{(b)}} f_{y_{qt}}^{(b)} \tag{5.10.1-11}$$

$$C_M^{(b)} = \frac{1}{A\bar{c}} \sum_{t=1}^{NSBE^{(b)}} \left[- f_{z_{qt}}^{(b)} (\xi_t^{(b)} - x_{LE}^{(b)}) + m_{z_{qt}}^{(b)} \right] \tag{5.10.1-12}$$

$$C_N^{(b)} = \frac{1}{A\bar{c}} \sum_{t=1}^{NSBE^{(b)}} \left[- f_{y_{qt}} (\xi_t^{(b)} - x_{LE}^{(b)}) + m_{y_{qt}}^{(b)} \right] \tag{5.10.1-13}$$

where

b is the index of the slender body,

$NSBE^{(b)}$ is the number of elements in slender body 'b'

$\xi_t^{(b)}$ is the x-coordinate of element 't' in slender body 'b', and

$x_{LE}^{(b)}$ is the leading edge x-coordinate of slender body 'b'.

The total lift and moment coefficients, including body effect are defined as follows:

$$C_{Z_q} = (1 + \delta)\left\{\frac{1}{A}\sum_{j=1}^{NSTRIP} G_j 2e_j c_j c_{n_{qj}} \cos\gamma_j + \sum_{b=1}^{NB} g^{(b)} C_{Z_q}^{(b)}\right\} \qquad (5.10.1\text{-}14)$$

$$C_{Y_q} = (1 - \delta)\left\{\frac{1}{A}\sum_{j=1}^{NSTRIP} G_j 2e_j c_j c_{n_{qj}} \sin\gamma_j + \sum_{b=1}^{NB} g^{(b)} C_{Y_q}^{(b)}\right\} \qquad (5.10.1\text{-}15)$$

$$C_M = (1 + \delta)\left\{\frac{1}{A\bar{c}}\sum_{j=1}^{NSTRIP} G_j \left[c^2 c_{m_{qj}} - c c_{n_{qj}}(\xi 14_j - XM)\right] 2e_j \cos\gamma_j + \sum_{b=1}^{NB} g^{(b)}[C_M^{(b)} - C_{Z_q}^{(b)}(x_{LE}^{(b)} - XM)/\bar{c}]\right\} \qquad (5.10.1\text{-}16)$$

$$C_N = (1 - \delta)\left\{\frac{1}{A\bar{c}}\sum_{j=1}^{NSTRIP} - G_j [c^2 c_{m_{qj}} - c c_{n_{qj}}(\xi 14_j - XM)] 2e_j \sin\gamma_j + \sum_{b=1}^{NB} g^{(b)}[C_N^{(b)} - C_{Y_q}^{(b)}(x_{LE}^{(b)} - XM)/\bar{c}]\right\} \qquad (5.10.1\text{-}17)$$

and

$$C_\ell = (1 - \delta)\,\frac{1}{2s}\left\{\frac{1}{A}\sum_{j=1}^{NSTRIP} G_j c c_{n_{qj}}(y_j \cos\gamma_j + z_j \sin\gamma_j) 2e_j + \sum_{b=1}^{NB} g^{(b)} C_{Z_q}^{(b)} y_c^{(b)} + \sum_{b=1}^{NB} g^{(b)} C_{Y_q}^{(b)} z_c^{(b)}\right\} \qquad (5.10.1\text{-}18)$$

where

A = reference area

XM = moment axis

δ = symmetry flag

$\left.\begin{array}{l} y_j \\ z_y \end{array}\right\}$ = y- and z-coordinates of the centerline of strip 'j'

$\left.\begin{array}{l} y_c^{(b)} \\ z_c^{(b)} \end{array}\right\}$ = y- and z-coordinates of the centerline of slender body 'b'

$$G_j = \begin{cases} 1/2 \text{ if } y_j = 0 \text{ and } \cos\gamma_j = 0 \text{ and } \delta \neq 0 \\ 1 \text{ otherwise} \end{cases} \qquad (5.10.1\text{-}19)$$

and

$$g^{(b)} = \begin{cases} 1/2 \text{ if } y_c^{(b)} = 0 \\ 1 \text{ otherwise} \end{cases} \qquad (5.10.1\text{-}20)$$

5.10 Segment 11

5.11.1 <u>SUBROUTINE GENF</u> (NMODE, NSTRIP, NW, NEWBFM, IMODE, AA, NAI, NPR2,IBFS)

<u>Functional Description</u>

Subroutine GENF computes generalized forces for all pressure and deflection modes according to either of the two definitions given in Equations (5.11.1-1) (AGARD definition) and(5.11.1-2) (conventional generalized forces) depending on the setting of the input flag NPR2 — 1 for AGARD forces, 2 otherwise. It also prints the pressure coefficients 'ΔC_p' for all boxes of all lifting surface panels along with the panel, strip and box number, and the fractional chordwise locations (x/c) for the 'ΔC_p'. Note that Subroutine GENF can be bypassed through MAIN by specifying NPR2 = 0 (input) whenever generalized forces and pressures are not desired.

<u>Input Output Variables</u>

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
NMODE NSTRIP NW NEWBFM		IN	ARG	See AERO, Sec. 5.10.1
IMODE (2,150,3)				Modal input array, IA(2,150,3) in Subroutine RDMODE; see Sec. 5.4.1
AA(2, 150,3)				Floating point equivalent of the array IMODE
NAI(3), NA(3)				See RDMODE, Sec. 5.4.1
NPR2				Control flag; 1 for AGARD generalized forces, 2 for conventional generalized forces
FZ(200) FY(200) MZ(200) MY(200)		IN	Tape BFSMAT	See Sec. 3.2
DCP(500)	ΔC_p	IN	Tape NW	See AERO, Sec. 5.10.1

MNEMONIC	SYMBOL	IN/OUT	SOURCE	DESCRIPTION
N8	N8	IN	GENF	One set of modal input for panel — see Sec. 5.4.1
N	n			
M	m			
AMODE	$a_{i_{nm}}$			
X40C	$x4_k/\bar{c}$			See Equation (5.11.1-4)
Y40C	$\tilde{y}4_k/\bar{c}$			
DELA	ΔA_k			Area of box 'k'
HQA(500)				See Equations(5.11.1-11 through -13)
DHQ(100)				See Equations(5.11.1-14 and -15)
BMODE	az_{i_n} ay_{i_n}			One modal coefficient for body; see Sec. 5.4.1
QW(50)	QW_{ij}	OUT	GENF	One row of each of the generalized force components for all pressure modes and one deflection mode — see Equations (5.11.1-3, -5 and -8)
QZ(50)	QZ_{ij}			
QY(50)	QY_{ij}			
QIJ(50)	Q_{ij}			See Equation (5.11.1-1 and -2)

Calling Subroutine MAIN

Common Blocks Blank Common Block

Equations

AGARD definition

$$Q_{ij} = \frac{-1}{2s^3}(QW_{ij} + QZ_{ij} + QY_{ij}) \qquad (5.11.1\text{-}1)$$

Conventional definition

$$Q_{ij} = \frac{1}{A\bar{c}}(QW_{ij} + QZ_{ij} + QY_{ij}) \qquad (5.11.1\text{-}2)$$

The Q_{ij} are computed for all deflection modes 'i' and all pressure modes 'j', where i and j run from 1 through NMODE.

The three components of the Q_{ij} are computed by Subroutine GENF as follows:

A. Lifting Surface Contribution

$$QW_{ij} = \sum_{k=1}^{NBOX} G\Delta C_{p_{kj}} h_{ki} \Delta A_k \qquad (5.11.1\text{-}3)$$

where

G_k = 1 if y-coordinate of centerline of strip is 0 and $\cos \gamma_{strip} = 0$
2 otherwise

$\Delta C_{p_{kj}}$ is the pressure coefficient for box 'k' for pressure mode 'j'

h_{ki} is the modal deflection of box 'k' at the 1/4-chord x-coordinate of the panel box centerline, '$x4_k$', defined by

$$h_{ki} = \bar{c} \sum_{n=0}^{5} \sum_{m=0}^{5} a_{i_{nm}} \left(\frac{x4_k}{\bar{c}}\right)^n \left(\frac{\tilde{y}4_k}{\bar{c}}\right)^m \qquad (5.11.1\text{-}4)$$

where

$$\tilde{y}4_k = \sqrt{(y_\ell - (N8)Y1^{(p)})^2 + (z_\ell - (N8)Z1^{(b)})^2}$$

y_ℓ and z_ℓ are the y- and z-coordinates of the centerline of strip 'ℓ', and $Y1^{(p)}$ and $Z1^{(p)}$ are y- and z-coordinates of the inboard edge of panel 'p', and ΔA_k = area of box 'k'.

The variables $a_{i_{nm}}$ and N8 are modal input data; see Section 1.2.

B. Slender Body Contribution

The QZ_{ij} and QY_{ij} in Equations (5.11.1-1 and -2) denote the z- and y-components of the slender contribution to the total generalized forces.

$$QZ_{ij} = \sum_{b=1}^{NB} g^{(b)} \sum_{t=1}^{NSBE^{(b)}} \left(f_{z_{ij}}^{(b)} h_{z_{ti}}^{(b)} + m_{z_{tj}} \frac{dh_{z_{ti}}^{(b)}}{dx} \right) \qquad (5.11.1\text{-}5)$$

where

$$h_{z_{ti}}^{(b)} = \bar{c}\sum_{n=0}^{5} az_{i_n}\left(\frac{xSB_t^{(b)}}{\bar{c}}\right)^n \qquad (5.11.1\text{-}6)$$

$$\frac{dh_{z_{ti}}}{dx} = \sum_{n=0}^{5} naz_{i_n}\left(\frac{xSB_t^{(b)}}{\bar{c}}\right)^{n-1} \qquad (5.11.1\text{-}7)$$

and

$$QY_{ij} = \sum_{b=1}^{NB} g^{(b)} \sum_{t=1}^{NSBE^{(b)}} \left(f_{y_{tj}} h_{y_{ti}} + m_{y_{tj}} \frac{dh_{y_{ti}}^{(b)}}{dx}\right) \qquad (5.11.1\text{-}8)$$

where

$$h_{y_{ti}}^{(b)} = \bar{c}\sum_{n=0}^{5} ay_{i_n}\left(\frac{xSB_t^{(b)}}{\bar{c}}\right)^n \qquad (5.11.1\text{-}9$$

$$\frac{dh_{y_{ti}}^{(b)}}{dx} = \sum_{n=0}^{5} nay_{i_n}\left(\frac{xSB_t^{(b)}}{\bar{c}}\right)^{n-1} \qquad (5.11.1\text{-}10)$$

and where the variables ai_{nm}, azi_n, ayi_n and $N\ell$ are modal input data (see Section 1.2); $xSB_t^{(b)}$ is the x-coordinate of the slender body element midpoint for element 't' of slender body 'b', and

$$g^{(b)} = \begin{cases} 1 & \text{if} \quad y_c^{(b)} = 0 \\ 2 & \text{if} \quad y_c^{(b)} \neq 0 \end{cases}$$

Note that, to facilitate programming, one array, HQA(500) is generated in Subroutine GENF for all boxes and all slender body elements, defined as

$$HQA(K) = Gh_k \Delta A_k, \qquad k = 1, NBOX \qquad (5.11.1\text{-}11)$$

and h_k is defined in Equation 5.11.1-4 as

$$HQA(KZ) = g^{(b)} h_{kz}, \qquad kz = kz_1, kz_2 \qquad (5.11.1\text{-}12)$$

where

$$ky_1 = kz_2 + 1, \qquad ky_2 = kz_2 + \sum_{b=1}^{NBY} NSBE^{(b)}$$

and h_{ky} is given in Equation (5.11.1-9).

Also, one array, DHQ(100), is generated to contain all the dh_z/dx and dh_y/dx in Equations (5.11.1-7) and (5.11.1-10) as follows:

$$DHQ(\ell z) = g^{(b)} \frac{dh_{z_{\ell z}}}{dx} , \qquad \ell z = \ell z_1, \ell z_2 \tag{5.11.1-14}$$

where

$$\ell z_1 = 1$$

and

$$\ell z_2 = \sum_{b=1}^{NBZ} NSBE^{(b)}$$

and

$$DHQ(\ell y) = g^{(b)} \frac{dh_{y_{\ell y}}}{dx}, \qquad \ell y = \ell y_1, \ell y_2, \tag{5.11.1-15}$$

where

$$\ell y_1 = \ell z_2 + 1$$

and

$$\ell y_2 = \ell z_2 + \sum_{b=1}^{NBY} NSBE^{(b)}$$

6.0 PROGRAM LISTING

```
C   SEGMENT  1  --  MAIN, ATAN3, DZY, FLLO, FMMM, IDF1, IDF2, INCRD,
C   REWND, MWREC, SNPDF, SUBI, TIME, TKF4, TVOR, WRTFMF, WRTFMU
C    ***                                                                MAIN0010
C      SUBSONIC UNSTEADY AERODYNAMICS FOR GENERAL CONFIGURATIONS        MAIN0020
C      PART 2  --  UNSTEADY IMAGE METHOD FOR GENERAL LIFTING SURFACE -  MAIN0030
C      BODY  COMBINATIONS                                               MAIN0040
C      METHOD CALCULATES LIFTING PRESSURE, SPAN LOADS AND MOMENTS,      MAIN0050
C      GENERALIZED FORCES, AND OTHER AERODYNAMIC CHARACTERISTICS FOR    MAIN0060
C      GENERAL LIFTING SURFACE - BODY COMBINATIONS.  STEADY AND UNSTEADY MAIN0070
C      FLOWS ARE CONSIDERED. BODIES MAY HAVE ELLIPTIC CROSS SECTIONS.   MAIN0080
C      LIFTING SURFACES MAY HAVE ARBITRARY PLANFORM AND DIHEDRAL.       MAIN0090
C      CONTROL SURFACES ARE HANDLED. MODES OF MOTION INPUT USING        MAIN0100
C      POLYNOMIALS. PROGRAM IS LARGE ENOUGH TO HANDLE ENTIRE AIRCRAFT.  MAIN0110
C    ***                                                                MAIN0120
      COMMON      NP,NTP,NB,NBZ,NBY,NTZ,NTY,FMACH,REFA,REFS,REFC,ND,NE, MAIN0130
     1           NBEA(10,2),NSBEA(10),NCARAY(100),NSARAY(100),NBARAY(100)MAIN0140
     2           ,ZA(10),YB(10),XLE(10),XTE(10),ARB(10),AVR(10),AO(200), MAIN0150
     3           AOP(200),XIS1(200),XIS2(200),XIJ(100),CS(210),YS(210), MAIN0160
     4           ZS(210),EE(200),SG(200),CG(200),X(500),XIC(500),KR,XM, MAIN0170
     5           DELX(500),XLAM(500),RIA(100),TH1A(100),TH2A(100),NFL(10)MAIN0180
     6           ,IFLA(30,2),NT12(10,2),NAS(100),NASB(200),YIN(100),    MAIN0190
     7           ZIN(100),DETA(50),DZET(50),NOBODY(50),NPANEL(50),NTO   MAIN0200
     8           ,NTZS,NTYS                                             MAIN0210
      REAL        KR                                                    MAIN0220
      DIMENSION   FREQ(10),HOR(15)                                      MAIN0230
      DIMENSION   IA(2,150,3),    NA(3)                                 MAIN0240
      COMPLEX     DT(500),RHS(100),WORK(10000)                          MAIN0260
      DIMENSION   NTAPE(20), NBEA1(10)                                  MAIN0270
      DATA        NTAPE/1, 2, 3, 4, 5, 6, 7, 8, 9,10,11,12, 8*0/        MAIN0280
      DATA        STOP/4HSTOP/                                          MAIN0290
   10 FORMAT   (1H0,15I5)                                               MAIN0300
   20 FORMAT   (15A4, I10)                                              MAIN0310
   30 FORMAT   (1H1,34X,15A4//)                                         MAIN0320
   40 FORMAT   (6F10.0)                                                 MAIN0330
   50 FORMAT   (6F20.6)                                                 MAIN0340
   60 FORMAT   (5F10.0, 5I2, 2I3, 6I1)                                  MAIN0350
   70 FORMAT   (1H0,30X,36H**  ARRAY OF REDUCED FREQUENCIES  **/ )      MAIN0360
   80 FORMAT   (1H0,5X,17HREFERENCE CHORD =,4X,F15.5    /6X,21HREFERENCE MAIN0370
     1SEMI-SPAN =, F15.5/6X,16HREFERENCE AREA =,5X,F15.5/6X,13HMACH NUMBMAIN0380
     2ER =,9X,F15.5/6X,13HMOMENT AXIS =,8X,F15.5/6X,27HSYMMETRY FLAGS  *MAIN0390
     3*  DELTA =,I3,11H  EPSILON =, I3/6X,18HNUMBER OF PANELS =, I4/6X, MAIN0400
     4  18HNUMBER OF BODIES =,I4//)                                     MAIN0410
   90 FORMAT   (1H0,25X,6F12.6/)                                        MAIN0420
  100 FORMAT   (1H0//10X, 65H***   BODY FORCES ARE COMPUTED ACCORDING TO MAIN0430
     1  OPTION NO.  1    ***//)                                         MAIN0440
  102 FORMAT   (1H0///25X, 14,44H  SLENDER BODY ELEMENT FORCES  **  MODE MAIN0442
     * NO., I4// 20X, 11HELEMENT NO., 12X, 2HFZ, 30X, 2HFY/)            MAIN0444
  104 FORMAT  (1H0, 24X, I4, 4E16.8)                                    MAIN0446
  110 CONTINUE                                                          MAIN0450
      CALL        TIME(0)                                               MAIN0460
      IERROR = 0                                                        MAIN0470
      NUMBR  = 0                                                        MAIN0475
C   ***    IERROR  IS AN ERROR FLAG USED IN SUBROUTINES   ROMODE,       MAIN0480
C          SB   AND     BFM     --       IF ANY OF THESE SUBROUTINES    MAIN0490
C          DETECTS   DATA ERROR ,  THE FLAG IS SET TO  1   --   AND,    MAIN0500
C          UPON EXIT FROM THIS SUBROUTINE  AN ERROR CHECK IS MADE --    MAIN0510
C          IF  IERROR IS NOT  ZERO  ,  THE CASE IS TERMINATED  AND      MAIN0520
C          THE PROGRAM PROCEEDS WITH THE EXECUTION OF THE NEXT CASE     MAIN0530
      NWORK   = 10000                                                   MAIN0540
      NOZ     = NTAPE(1)                                                MAIN0550
      NOY     = NTAPE(2)                                                MAIN0560
      MTAPE   = NTAPE(2)                                                MAIN0570
      ITAPE   = NTAPE(8)                                                MAIN0580
      NPTAP   = NTAPE(3)                                                MAIN0590
      NIN     = NTAPE(5)                                                MAIN0600
      NOUT    = NTAPE(6)                                                MAIN0610
      NPSTAP = NTAPE(9)                                                 MAIN0620
      NM     = NTAPE(1)                                                 MAIN0630
      NG     = NTAPE(2)                                                 MAIN0640
      NOW    = NTAPE(3)                                                 MAIN0650
      NWT    = NTAPE(4)                                                 MAIN0660
      NI     = NTAPE(10)                                                MAIN0670
      NBFM   = NTAPE(11)                                                MAIN0680
      NEWBFM = NTAPE(8)                                                 MAIN0685
      NW     = NPTAP                                                    MAIN0690
      READ  (5,20)  (HOR(I),  I=1,15),  IBFS                            MAIN0700
      IF (HOR(1).EQ.STOP)  STOP                                         MAIN0710
      WRITE (6,30)  (HOR(I), I=1,15)                                    MAIN0720
      IF  (IBFS . NE . 0)  WRITE  (6,100)                               MAIN0730
      READ  (5,50)  FMACH, REFA,REFS,REFC,XM,  ND,NE,NP,NB,NK,  MK1,MK2,MAIN0740
     1                N1,N2,N3,N4                                       MAIN0750
C   ***    N1  THROUGH  N4  ARE  CONTROL FLAGS                          MAIN0760
C   ***    N1  IS THE SOLUTION  PRINT FLAG  --                          MAIN0770
C          N1=0  MEANS  NO  PRINT  --                                   MAIN0780
C          N1=1  MEANS  ALL SOLUTIONS PRINTED                           MAIN0790
C   ***    N2 IS THE GENERALIZED FORCE FLAG                             MAIN0800
C          N2=0  MEANS     NO GENERALIZED FORCES                        MAIN0810
C          N2=1  MEANS  *AGARD*  GENERALIZED  FORCES                    MAIN0820
C          N2=2  MEANS  GENERALIZED FORCES --  USUAL DEFINITION         MAIN0830
C   ***    N3  IS THE MATRIX-PRINT-FLAG   --   N3=0  MEANS  NO DT-PRINT  MAIN0840
C          N3=1  MEANS DT-PRINT                                         MAIN0850
C   ***    N4  IS A PRINT FLAG USED IN  SUBROUTINES  BFM,  ROMODE,      MAIN0860
C          SB  AND HANDWT    --    N4=0  MEANS NO  EXTRA PRINT IN THESE  MAIN0870
C          SUBROUTINES   --   N4=1  MEANS  PRINT  ADDITIONAL INTERMEDIATEMAIN0880
C          RESULTS AND RELATED  INFORMATION IN  SUBROUTINE  BFM ONLY    MAIN0890
C          N4=2  MEANS DETAIL PRINT  IN ALL FOUR SUBROUTINES            MAIN0900
      BETA2  = 1.0 - FMACH**2                                           MAIN0910
      BETA   = SQRT( BETA2 )                                            MAIN0920
      READ  (5,40)  (FREQ(I),  I=1,NK)                                  MAIN0930
      WRITE (6,70)                                                      MAIN0940
      WRITE (6,90)  (FREQ(I),  I=1,NK)                                  MAIN0950
      WRITE (6,80)  REFC,REFS,REFA,FMACH,XM,ND,NE,NP,NB                 MAIN0960
      NPR1 = N1                                                         MAIN0970
      NPR2 = N2                                                         MAIN0980
      NPRINT = N3                                                       MAIN0990
      CALL        TIME(1)                                               MAIN1000
      CALL        DATA                                                  MAIN1010
      CALL        TIME(1)                                               MAIN1020
      NTOT = NTP+NTZ+NTY                                                MAIN1030
      NTIBE  = NTZ  +  NTY                                              MAIN1040
      NTSBE  = NTZS + NTYS                                              MAIN1050
      N         =     NTP   +  NTIBE                                    MAIN1060
      DO   120  I  =  1 , N                                             MAIN1070
      WORK (I)  =  CMPLX ( 0.0  ,  0.0  )                               MAIN1080
  120 CONTINUE                                                          MAIN1090
      NSTRIP = 0                                                        MAIN1100
      DO  130  L=1,NP                                                   MAIN1110
      NSTRIP = NSTRIP+NSARAY(L)                                         MAIN1120
  130 CONTINUE                                                          MAIN1130
  140 CONTINUE                                                          MAIN1140
C                                                                       MAIN1150
      KD    = 2*NWORK                                                   MAIN1160
      NR    = NTOT                                                      MAIN1170
      DO  280  J=1,NK                                                   MAIN1180
      KR    = FREQ(J)                                                   MAIN1190
      CALL        TIME(1)                                               MAIN1200
      CALL        GEND(NPRINT,NTAPE,WORK)                               MAIN1210
      CALL        TIME(1)                                               MAIN1220
```

```
      N4S   = N4                                                        MAIN1230
      IF  (N4.NE.2)  N4 = 0                                             MAIN1240
      CALL        RDMODE(IA,  NA,  NIN,  NMT, NOUT, N4, IERROR)         MAIN1250
      IF  (IERROR.NE.0)  GO TO  280                                     MAIN1260
      CALL        TIME(1)                                               MAIN1270
C                                                                       MAIN1280
      NMODE  = NMT                                                      MAIN1290
      IF  (NB.EQ.0)  GO TO  150                                         MAIN1300
      CALL        TIME(1)                                               MAIN1310
      CALL        SB( IA, NMT, NA(3), NA(2), WORK, NWORK, N4        )   MAIN1320
C     IF  (IERROR.NE.0)  GO TO  280                                     MAIN1330
  150 CONTINUE                                                          MAIN1340
      CALL        TIME(1)                                               MAIN1350
      CALL        WANDWT(IA, IA, NSARAY, NBARAY, X, YS, YIN, ZS, ZIN,   MAIN1360
     1     NR, REFC, KR, NP, NTP, NMT, NA(1), NDW, NWT, NOUT,           MAIN1370
     2     WORK(1), WORK(NTP+1), NB,  N4)                               MAIN1380
      IF  (IERROR.NE.0)  GO TO  280                                     MAIN1390
      CALL        TIME(1)                                               MAIN1400
      NIM      =    NTIBE   + NTSBE  + NTP   + 1                        MAIN1410
      NFC      =  NIM + ( NTP + 1)/2 + NTIBE + MAX0(NTP+NTIBE,NTSBE)    MAIN1420
      NWKR = NWORK  -  NFC                                              MAIN1430
C                                                                       MAIN1440
C                                                                       MAIN1450
  160 CONTINUE                                                          MAIN1460
      REWIND NMT                                                        MAIN1470
      JX2   = 0                                                         MAIN1480
      DO 170  JJ=1,NMODE                                                MAIN1490
      JX1   = JX2+1                                                     MAIN1500
      JX2   = JX2+NTOT                                                  MAIN1510
      READ  (NMT )  (WORK(JX),  JX=JX1,JX2)                             MAIN1520
  170 CONTINUE                                                          MAIN1530
  180 CONTINUE                                                          MAIN1540
      REWIND  NI                                                        MAIN1550
      REWIND  NM                                                        MAIN1560
      REWIND  NO                                                        MAIN1570
      REWIND  NW                                                        MAIN1580
      REWIND  ITAPE                                                     MAIN1590
      DO  200  IX=1,NTOT                                                MAIN1600
      JM    = 0                                                         MAIN1610
      READ  (ITAPE)  (DT(II),  II=1,NTOT)                               MAIN1620
      DO 190  JX=1,NMODE                                                MAIN1630
      JM    = JM+1                                                      MAIN1640
      JJ    = (JM-1)*NTOT + IX                                          MAIN1650
      RHS(JX) =  WORK(JJ)                                               MAIN1660
      IF  (IX.GE.NK1. AND .IX.LE.NK2)    RHS(JX) = (0.,                 MAIN1670
  190 CONTINUE                                                          MAIN1680
      WRITE  (NO)  ((DT(K),  K=1,NTOT), (RHS(JK),  JK=1,NMODE))         MAIN1690
  200 CONTINUE                                                          MAIN1700
      REWIND  NI                                                        MAIN1710
      CALL        TIME(1)                                               MAIN1720
      CALL        SOLVIT(WORK,WORK,NTOT,NMODE,KD,NI,NM,NO,NW,NPR1)      MAIN1730
      CALL        TIME(1)                                               MAIN1740
      IF  (NTSBE . EQ . 0)  GO TO  270                                  MAIN1750
      REWIND   NBFM                                                     MAIN1760
      NS    = N4S                                                       MAIN1770
      IF  (IBFS . NE . 0)  GO TO  210                                   MAIN1780
      CALL        BFM( WORK, WORK(NIM), WORK(NFC), NWKR,        NPTAP,  MAIN1790
     1     NPSTAP, NBFM, NOUT, NMT, NS, IERROR, IBFS)                   MAIN1800
      IF  (IERROR.NE.0)  GO TO  280                                     MAIN1810
      CALL        TIME(1)                                               MAIN1820
      GO TO  270                                                        MAIN1830
  210 CONTINUE                                                          MAIN1840
      NSBETO = 0                                                        MAIN1850
      DO  220  I=1,NB                                                   MAIN1860
      NBEA1(I) = NBEA(I , 1)                                            MAIN1865
  220 NSBETO = NSBETO + NSBEA(I)                                        MAIN1870
      LENGTH = NTOT + NTZS + NTYS                                       MAIN1875
      CALL        BFSMAT( ND    ,   NE    ,   NR    ,   NP    ,   NTP   ,MAIN1880
     1 LENGTH, NTOT  ,   NBFM  ,   NS    ,   NAS   ,   FMACH ,   YR    ,MAIN1890
     2         ZB    ,   YS    ,   ZS    ,   X     ,   DELX  ,   FE    ,MAIN1900
     3         XIC   ,   SG    ,   CG    ,   ARB   ,   RIA   ,   NBEA1 ,MAIN1910
     4NBEA(1,2),NASB ,   NSARAY,   NCARAY,   WORK  ,   AVR   ,   REFC  ,MAIN1920
     5         AO    ,   XIS1  ,   XIS2  ,   KR    ,   NSBEA ,   IBFS  )MAIN1930
      CALL        TIME(1)                                               MAIN1940
C                                                                       MAIN1950
C   ***   THE FOLLOWING IS A MATRIX MULTIPLICATION  TO OBTAIN THE  Z-   MAIN1960
C         AND  Y-FORCES ON ALL SLENDER BODY ELEMENTS AND FOR ALL        MAIN1970
C         MODES  USING THE OUTPUT MATRIX OF THE NEW  SUBROUTINE  BFSMAT MAIN1980
C         AND THE SOLUTION MATRIX  OBTAINED BY  SUBROUTINE  SOLVIT      MAIN1990
C                                                                       MAIN2000
C                                                                       MAIN2010
      REWIND  NW                                                        MAIN2020
      REWIND  NPSTAP                                                    MAIN2025
C                                                                       MAIN2030
      REWIND  NEWBFM                                                    MAIN2040
      NSBET2 = 2 * NSBETO                                               MAIN2045
      NBEG    = 1000                                                    MAIN2050
      NBEG1   = NBEG + 1                                                MAIN2060
      NBEG2   = NTOT + 1                                                MAIN2070
      NUMBR   = 0                                                       MAIN2080
      DO  260  J=1,NMODE                                                MAIN2090
      REWIND  NBFM                                                      MAIN2092
      WRITE  (6,102)  NSBETO, J                                         MAIN2093
      DO  230  IM=1,NWORK                                               MAIN2094
  230 WORK(IM) = 0.0                                                    MAIN2096
      IFLAG   = 1                                                       MAIN2100
C                                                                       MAIN2110
C   ***   READ ONE COLUMN OF SOLUTIONS  FROM TAPE  NW                   MAIN2120
C                                                                       MAIN2130
      CALL        RWREC(IFLAG,  NW , DT(  1  ), NTOT  , NUMBR )         MAIN2140
C                                                                       MAIN2150
C   ***   APPEND THE ARRAY CONTAINING ONE COLUMN OF SOLUTIONS  BY THE   MAIN2160
C         CORRESPONDING COLUMN OF   -Z-DELTA-A  AND CP-Y-DELTA-A  **    MAIN2170
C         THESE ARE READ FROM TAPE  NPSTAP                              MAIN2180
C                                                                       MAIN2190
      CALL        RWREC(IFLAG, NPSTAP, DT(NBEG2),  NTSBE, NUMBR )       MAIN2200
      DO  250  I =1,NSBET2                                              MAIN2210
      IFLAG   = 1                                                       MAIN2220
C                                                                       MAIN2230
      CALL        RWREC(IFLAG, NBFM , WORK(1), LENGTH, NUMBR)           MAIN2240
                                                                        MAIN2250
      NBW   = NBEG + I                                                  MAIN2260
      DO  240  K=1, LENGTH                                              MAIN2270
      WORK(NBW) = WORK(NBW) + WORK(K) * DT(K)                           MAIN2280
  240 CONTINUE                                                          MAIN2290
  250 CONTINUE                                                          MAIN2300
      IFLAG   = 0                                                       MAIN2310
      CALL        RWREC(IFLAG, NEWBFM, WORK(NBEG1), NSBET2, NUMBR)      MAIN2320
      IWY     = NBEG1 - 1                                               MAIN2330
      DO  255  K = 1, NSBETO                                            MAIN2340
      IWZ     = IWY + 1                                                 MAIN2350
      IWY     = IWZ + 1                                                 MAIN2360
      WRITE  (6,104)  K, WORK(IWZ), WORK(IWY)                           MAIN2370
  255 CONTINUE                                                          MAIN2380
  260 CONTINUE                                                          MAIN2390
C                                                                       MAIN2400
C   ***   MATRIX MULTIPLICATION COMPLETED  --  RESULT SAVED ON TAPE     MAIN2410
C         UNIT  'NEWBFM'                                                MAIN2420
C                                                                       MAIN2430
  270 CONTINUE                                                          MAIN2440
```

```
      CALL       AERO(NMODE,NSTRIP,NW,  NBFM,NEWBFM,IBFS)                   MAIN2450
      CALL       TIME(1)                                                    MAIN2460
      IF (NPR2.EQ.0)  GO TO  280                                            MAIN2470
      CALL       GENF(NMODE,NSTRIP,NW,NEWBFM,IA    ,IA,NA ,NPR2,IBFS)       MAIN2480
      CALL       TIME(1)                                                    MAIN2490
  280 CONTINUE                                                              MAIN2500
      GO TO 110                                                             MAIN2510
      END                                                                   MAIN2520


      SUBROUTINE ATAN3(Y,X,T)                                               ATAN0010
C   ***   COMPUTES  T=ATAN(Y/X)                                             ATAN0015
      PI = 3.141593                                                         ATAN0020
      PI2 = PI*2.0                                                          ATAN0030
      IF(X.EQ.0.) T=PI/2.                                                   ATAN0040
      IF(X.EQ.0.) GO TO 2                                                   ATAN0050
      IF(Y.EQ.0.)T=0.                                                       ATAN0060
      IF(Y.EQ.0.0) GO TO 2                                                  ATAN0070
      A = Y/X                                                               ATAN0080
      T = ATAN(A)                                                           ATAN0090
    2 CONTINUE                                                              ATAN0100
      IF((Y.GE.0.).AND.(X.EQ.0.)) GO TO 1                                   ATAN0110
      IF ((Y.EQ.0.).AND.(X.GT.0.)) T=0.                                     ATAN0120
      IF ((Y.EQ.0.).AND.(X.GT.0.)) GO TO 1                                  ATAN0130
      IF((Y.GE.0.).AND.(X.LE.0.))T = PI + T                                 ATAN0140
      IF((Y.GE.0.).AND.(X.LE.0.))GO TO 1                                    ATAN0150
      IF((Y.LE.0.).AND.(X.LE.0.))T = PI + T                                 ATAN0160
      IF((Y.LE.0.).AND.(X.LE.0.))GO TO 1                                    ATAN0170
      IF((Y.LE.0.).AND.(X.GE.0.))T = PI2+ T                                 ATAN0180
      IF((Y.LE.0.).AND.(X.GE.0.))GO TO 1                                    ATAN0190
    1 CONTINUE                                                              ATAN0200
      RETURN                                                                ATAN0210
      END                                                                   ATAN0220


CDZY            CALCULATION OF THE DZ AND DY MATRICES                       DZY00010
C               USED IN SLENDER BODY FLOW                                   DZY00020
C                                                                           DZY00030
      SUBROUTINE  DZY  ( X, Y, Z, SGR, CGR, XI1, XI2, ETA, ZETA, AR, AO,DZY00040
     *    KR     , CBAR    , BETA    , FMACH   ,                            DZY00050
     *     IDZDY,    DZDYR,    DZDYI,     IPRNT    )                        DZY00060
C                                                                           DZY00070
C *                                                                       * DZY00080
C *      ********* ***************************************************** DZY00090
C *                                                                       * DZY00100
C *       X          X- COORDINATE OF THE RECEIVING POINT                 * DZY00110
C *       Y          Y - COORDINATE OF THE RECEIVING POINT                * DZY00120
C *       Z          Z - COORDINATE OF THE RECEIVING POINT                * DZY00130
C *       SGR        SINE OF THE RECEIVING POINT DIHEDRAL ANGLE           * DZY00140
C *       CGR        COSINE OF RECEIVING POINT DIHEDRAL ANGLE             * DZY00150
C *       XI1                                                             * DZY00160
C *       XI2                                                             * DZY00170
C *       ETA                                                             * DZY00180
C *       ZETA                                                            * DZY00190
C *       AR         ASPECT RATIO OF THE SENDING BODY                     * DZY00200
C *       AO         RADIUS OF THE SENDING BODY                           * DZY00210
C *       KR         REDUCED FREQUENCY                                    * DZY00220
C *       CBAR       REFERENCE CHORD LENGTH                               * DZY00230
C *       BETA       SQRT(1.0-M**2)                                       * DZY00240
C *       FMACH      MACH NUMBER                                          * DZY00250
C *       IDZDY      FLAG INDICATING WHETHER DZ OR DY IS TO BE            * DZY00260
C *                  CALCULATED.  =0  DZ,    OTHERWISE DY                 * DZY00270
C *       DZDYR      REAL PART OF DZ OR DY                                * DZY00280
C *       DZDYI      IMAGINARY PART OF DZ OR DY                           * DZY00290
C *                                                                       * DZY00300
C *      ********* ***************************************************** DZY00310
C *                                                                       * DZY00320
      DATA  PI16 / 50.265482 /                                              DZY00330
C                                                                           DZY00340
      REAL   K01PR, K01PI, K02PR, K02PI, K01MR, K01MI, K02MR, K02MI         DZY00350
C                                                                           DZY00360
C          THE COMPLEX NUMBERS IN THIS ROUTINE ARE TREATED SEPERATLY AS     DZY00370
C          THE REAL PART,  NAME APPENDED BY AN  -R- ,  AND THE *            DZY00380
C          IMAGINARY PART, NAME APPENDED BY AN  -I- .                       DZY00390
C                                                                           DZY00400
      DATA     NPOT  /6/                                                    DZY00410
      IF  (  IPRNT  .NE.  0    )     WRITE (  NPOT , 100 )                  DZY00420
     *    X, Y, Z, SGR,  CGR,   XI1,  XI2,  ETA,  ZETA ,                    DZY00430
     *   AR, AO, KR, IDZDY                                                  DZY00440
  100 FORMAT ( 23H DZDY INPUT-- X, Y, Z, SGR, CGR, XI1, XI2, ETA, ZETA, DZY00450
     *AR, AO, KR, IDZDY        /  10E12.4 / 2E12.4, I4  )                   DZY00460
C                                                                           DZY00470
      E       =     AO * SQRT ( ABS( 1.0 - AR**2) ) / 2.0                   DZY00480
      X01    =  X   - XI1                                                   DZY00490
      X02    =  X   - XI2                                                   DZY00500
C                                                                           DZY00510
C                               CHECK ON INPUT FLAG                         DZY00520
C                               = 0  DZ ,   = 1  DY                         DZY00530
C                                                                           DZY00540
      IF ( IDZDY .EQ. 1 )  GO TO 200                                        DZY00550
C                                          * *      * *                     DZY00560
C                                          *   D Z   *                      DZY00570
C                                          * *      * *                     DZY00580
      SGS    =  0.0                                                         DZY00590
      CGS    =  1.0                                                         DZY00600
      IF ( AR .LT. 1.0 )  GO TO 400                                         DZY00610
C                                                                           DZY00620
      Z01        =  Z   - ( ZETA + E )                                      DZY00630
      Z02        =  Z   - ( ZETA - E )                                      DZY00640
      Y01        =  Y  -    ETA                                             DZY00650
      Y02        =  Y01                                                     DZY00660
      GO TO  300                                                            DZY00670
C                                          * *      * *                     DZY00680
C                                          *   D Y   *                      DZY00690
C                                          * *      * *                     DZY00700
  200 SGS        =  -1.0                                                    DZY00710
      CGS        =   0.0                                                    DZY00720
      IF ( AR .GT. 1.0 )  GO TO 400                                         DZY00730
C                                                                           DZY00740
      Z01        =  Z  -    ZETA                                            DZY00750
      Z02        =  Z01                                                     DZY00760
      Y01        =  Y   - ( ETA + E)                                        DZY00770
      Y02        =  Y   - ( ETA - E)                                        DZY00780
C     GO TO 300                        ****  DZ    AR    GE  1  ****        DZY00790
                                       ****  DY    AR    LE  1  ****        DZY00800
```

```
  300 CONTINUE                                                          DZY00810
      L          =     0                                                DZY00820
      ZO         =     Z - ZETA                                         DZY00830
      YO         =     Y -  ETA                                         DZY00840
C                                                                       DZY00850
      R1SQR      =     YO1**2  +  ZO1**2                                DZY00860
      R2SQR      =     YO2**2  +  ZO2**2                                DZY00870
      R1FOR      =     R1SQR**2                                         DZY00880
      R2FOR      =     R2SQR**2                                         DZY00890
C                                                                       DZY00900
      IF ( IPRNT .NE. 0 ) WRITE(NPOT,1000)XO1,XO2,YO1,YO2,ZO1,ZO2 ,     DZY00910
     *    SGR, CGR, SGS,                                                DZY00920
     *     CGS      , KR        , CBAR     , FMACH    , E         , L   DZY00930
C                                                                       DZY00940
      CALL     FLLD (XO1    , XO2    , YO1    , ZO1    , SGR    , CGR    , DZY00950
     *                SGS    , CGS    , KR      , CBAR   ,                DZY00960
     *                FMACH, E,  L, KO1PR   , KO1PI   , KO2PR   , KO2PI    ) DZY00970
C                                                                       DZY00980
      IF ( AR .NE. 1.0 )  GO TO 320                                     DZY00990
C          IDENTICAL RESULTS FROM FLLD, THEREFORE SKIP SECOND CALL      DZY01000
      KO1MR      =   KO1PR                                              DZY01010
      KO1MI      =   KO1PI                                              DZY01020
      KO2MR      =   KO2PR                                              DZY01030
      KO2MI      =   KO2PI                                              DZY01040
      GO TO 360                                                         DZY01050
  320 CONTINUE                                                          DZY01060
      CALL FLLD     (XO1    , XO2    , YO2    , ZO2    ,SGR    , CGR    , DZY01070
     *                SGS    , CGS    , KR     ,, CBAR   )                DZY01080
     *                FMACH, E,  L, KO1MR   , KO1MI   , KO2MR   , KO2MI    ) DZY01090
      IF ( IPRNT .NE. 0)                                                DZY01100
     * WRITE ( NPOT, 1000)  KO1PR, KO1PI,  KO2PR,  KO2PI,                DZY01110
     *                      KO1MR, KO1MI, KO2MR, KO2MI                  DZY01120
  360 CONTINUE                                                          DZY01130
      DZOYR   = 0.0                                                     DZY01140
      DZOYI   = 0.0                                                     DZY01150
      IF   (R1SQR.LE.0.0001. OR .R2SQR.LE.0.0001)  GO TO   370          DZY01160
C                REAL                                                   DZY01170
      DZOYR   = (   (KO1PR / R1SQR   + KO1MR / R2SQR )                  DZY01180
     *            + ( KO2PR / R1FOR   +  KO2MR / R2FOR ) ) / PI16       DZY01190
     *      * ( -1.0 )                                                  DZY01200
C                IMAGINARY                                              DZY01210
      DZOYI   = (   (KO1PI / R1SQR   + KO1MI / R2SQR )                  DZY01220
     *            + ( KO2PI / R1FOR   +  KO2MI / R2FOR ) ) / PI16       DZY01230
     *      * ( -1.0 )                                                  DZY01240
  370 CONTINUE                                                          DZY01250
      IF ( IPRNT .NE. 0 )  WRITE ( NPOT,1000) DZOYR,DZOYI               DZY01260
C                                                                       DZY01270
C                                                                       DZY01280
      RETURN                                                            DZY01290
C                               ****   DZ  -   AR  .LT.  1   ****       DZY01300
C                               ****   DY  -   AR  .GT.  1   ****       DZY01310
  400 SL1        =    0.0                                               DZY01320
      TL1        =    0.0                                               DZY01330
      SL2        =    0.0                                               DZY01340
      TL2        =    0.0                                               DZY01350
      CL1        =    1.0                                               DZY01360
      CL2        =    1.0                                               DZY01370
      E          =     1.732051 * E                                     DZY01380
      YO         =     Y - ETA                                          DZY01390
      ZO         =     Z - ZETA                                         DZY01400
C                                                                       DZY01410
      CALL  TVOR  ( SL1, CL1, TL1, SL2, CL2, TL2, SGS, CGS, SGR, CGR,   DZY01420
     *      XO1    , XO2    , YO     , ZO     , E        , BETA  , CBAR   , DZY01430
     *                FMACH , KR       ,                                DZY01440
     *      DZOYR,    DZOYI,     IPRNT    )                             DZY01450
      RETURN                                                            DZY01460
 1000 FORMAT  ( 5E20.3/ 5E20.8 / I20, 3E20.8, I10  )                    DZY01470
      END                                                               DZY01480


CFLLD             CALCULATION OF THE NUMERATOR OF A DOUBLET LINE OF FINITE FLL00010
C     LENGTH.     LIKE KERN, THERE ARE TWO OUTPUT COMPLEX VALUES REPRESENT-FLL00020
C                 ED BY FOUR REAL NUMBERS AND AN INPUT OPTION.          FLL00030
C                                                                       FLL00040
C                 WRITTEN BY D. H. LARSON, STRUCTURAL MECHANICS MDAC 11/70 FLL00050
C                                                                       FLL00060
      SUBROUTINE  FLLD  ( XO1, XO2, YO, ZO, SGR, CGR, SGS, CGS,  KR,    FLL00070
     *     CBAR  , FMACH , E      , L       , KO1R, KO1I  , KO2R  , KO2I  ) FLL00080
C                                                                       FLL00090
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * FLL00100
C *         XO1   -     X - XI1                                       * FLL00110
C *         XO2   -     X - XI2                                       * FLL00120
C *         YO    -     Y - ETA                                       * FLL00130
C *         ZO    -     Z - ZETA                                      * FLL00140
C *         SGR   -     SIN ( GAMMA-R)                                * FLL00150
C *         CGR   -     COS ( GAMMA-R)                                * FLL00160
C *         SGS   -     SIN ( GAMMA-S)                                * FLL00170
C *         CGS   -     COS ( GAMMA-S)                                * FLL00180
C *         KR    -     REDUCED FREQUENCY                             * FLL00190
C *         BR    -     REFERENCE LENGTH                              * FLL00200
C *         FMACH-      MACH NUMBER                                   * FLL00210
C *         E     -                                                   * FLL00220
C *         L     -     OPTION FLAG USED IN TKER                      * FLL00230
C *         KO1R -      REAL PART OF  KO1                             * FLL00240
C *         KO1I -      IMAGINARY PART OF KO1                         * FLL00250
C *         KO2R -      REAL PART OF  KO2                             * FLL00260
C *         KO2I -      IMAGINARY PART OF KO2                         * FLL00270
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * FLL00280
      COMMON  /KOS /  IND, KK1R,   KK1I,  KK2R,  KK2I,  R1             FLL00290
      COMMON  /DLM/  K10, K20, K1RT1,  K1IT1, K2RT2P, K2IT2P,  K1OT1,  FLL00300
     *                K2OT2P,  E2                                       FLL00310
      REAL       KR,  KK1R,   KK1I,  KK2R,  KK2I                        FLL00320
      REAL       KO1R,     KO1I ,     KO2R,       KO2I                  FLL00330
      REAL  K1OT1, K2OT2P, K1RT1, K10, K2IT2P, K20, K2RT2P,K1IT1        FLL00340
      COMPLEX  KO1, KO2, K1XI1, K1XI2, TEMP1, TEMP2, K2XI1, K2XI2       FLL00350
C                                                                       FLL00360
C              XO1 =  X - XI1  AND  XO2 = X - XI2, DELXI= XI2- XI1      FLL00370
      DELXI     =   XO1 - XO2                                           FLL00380
C                                                                       FLL00390
C                 FULL KERNEL FROM - TKER -                             FLL00400
      IND       =   0                                                   FLL00410
      KO1R      =   0.0                                                 FLL00420
      KO2R      =   0.0                                                 FLL00430
      E2        =   E **2                                               FLL00440
      T1        =     KR * DELXI / CBAR                                 FLL00450
      BR        =     CBAR / 2.0                                        FLL00460
```

```
      STI      =   SIN ( TI )
      CTI      =   COS ( TI )
      I        =   1
      XO       =   XO1
C
C
   10 CALL  TKER  ( XO    ,   YO    ,   ZO    ,   KR    ,   BR    ,
     *     SGR    ,   CGR   ,   SGS   ,   CGS   ,   RT1   ,   RT2   ,
     *     FMACH )
C
      GO TO  (  30, 40), I
   30 K1XI1    =  CMPLX ( KK1R, KK1I )
      K2XI1    =  CMPLX ( KK2R, KK2I )
      IF ( L  .EQ.  0 )  GO TO 35
      KO1R     =   KO1R   -  K1OT1
      KO2R     =   KO2R   -  K2OT2P
   35 CONTINUE
C
C          NOW GO CALCULATE FOR XI = XI2
C
      XO   =   XO2
      I    =   2
      GO TO  10
C
   40 K1XI2    =  CMPLX ( KK1R, KK1I )
      K2XI2    =  CMPLX ( KK2R, KK2I )
      IF ( L  .EQ. 0  )  GO TO 50
      KO1R     =   KO1R   +  K1OT1
      KO2R     =   KO2R   +  K2OT2P
   50 CONTINUE
C
      TEMP1    =  CMPLX ( CT1,  ST1  )
      TEMP2    =  CMPLX ( CT1, -ST1  )
C
C                DESIRED RESULTS  ( COMPLEX )
      KO1      =  K1XI1 * TEMP1  -  K1XI2 * TEMP2
      KO2      =     K2XI1 * TEMP1 - K2XI2 * TEMP2
C
C                CONVERT TO REAL AND IMAGINARY PARTS
      KO1R     =  REAL ( KO1 )  +  KO1R
      KO1I     = AIMAG  ( KO1 )
      KO2R     =  REAL ( KO2 )  +  KO2R
      KO2I     = AIMAG  ( KO2 )
      RETURN
      END


      SUBROUTINE  FWMW  (  ND,    NE,     SGS,    CGS ,    IRB,    AO     ,
     *     ARB     , XBLE     , XBTE     , YB       , ZB       , XS       ,
     *     YS      , ZS       , NAS      , NASB     , KR       , BETA2    ,
     *     CBAR    , AVR      , FWZ      , FWY      , MWZ      , MWY      ,
     *     IFI     , IPRNT    , IBFS     )
C   ***    CALCULATES THE EFFECT OF A DOUBLET PLUS ANY CONTRIBUTIONS
C          DUE TO IMAGES, SYMMETRY  AND GROUND EFFECT ON BODY
C *                                                                    *
C *        ********* ******************************************************
C *                                                                    *
C *       ND        SYMMETRY FLAG
C *       NE        GROUND EFFECTS FLAG                                 *
C *       SGS       SINE OF SENDING POINT DIHEDRAL ANGLE                *
C *       CGS       COSINE OF SENDING POINT DIHEDRAL ANGLE              *
C *       IRB       NUMBER OF THE RECEIVING BODY                        *
C *       AO        RADIUS OF THE BODY                                  *
C *       ARB       ARRAY OF RATIOS OF BODY AXIS                        *
C *       XBLE      LEADING  EDGE COORDINATE OF SLENDER BODY ELEMENT
C *       XBTE      TRAILING EDGE COORDINATE OF SLENDER BODY ELEMENT
C *       YB        ARRAY CONTAINING THE Y-COORDINATES OF THE           *
C *                 BODIES                                              *
C *       ZB        ARRAY CONTAINING THE Y-COORDINATES OF THE           *
C *                 BODIES                                              *
C *       XS        1/4-CHORD  X-COORDINATE  OF SLENDER BODY ELEMENT
C *       YS        Y-COORDINATE OF SENDING POINT                       *
C *       ZS        Z COORDINATE OF THE SENDING POINT                   *
C *       NAS       NUMBER OF ASSOCIATED BODIES                         *
C *       NASB      ARRAY CONTAINING THE ASSOCIATED BODY NOS.           *
C *       KR        REDUCED FREQUENCY                                   *
C *       BETA2     = 1 - ( MACH ** 2 )
C *       CBAR      REFERENCE CHORD LENGTH                              *
C *       AVR       ARRAY OF BODY RADII                                 *
C *       FWZ       OUTPUT Z-FORCE                                      *
C *       FWY       OUTPUT Y FORCE                                      *
C *       MWZ       OUTPUT Z MOMENT                                     *
C *       MWY       OUTPUT Y MOMENT                                     *
C *       IFI       FLAG INDICATING THE ORIENTATION OF THE              *
C *                 RECEIVING BODY                                      *
C *       IPRINT    PRINT FLAG
C *       IBFS      OPTION FLAG  -  0  TO SELECT  SUBROUTINE FMZY,
C *                                 1  TO SELECT  SUBROUTINE FZY2
C *                                                                    *
C *        ********* ******************************************************
C *                                                                    *
      DIMENSION     YB    (  1  ), ZB    (  1  ), NASB  (  1  ),
     *     AVR   (  1  ),IRB   (  1  )
      COMPLEX   FWZ,    FWY,    MWZ,    MWY
      REAL     MWZR,  MWZI,  MWYR,  MWYI,  KR
      DATA   NPOT  /6/
      FWZ     =   CMPLX  (0.0,  0.0)
      FWY     =   CMPLX  (0.0,  0.0)
      MWZ     =   CMPLX  (0.0,  0.0)
      MWY     =   CMPLX  (0.0,  0.0)
C
C
      IF( IPRNT  .NE. 0 )  WRITE ( NPOT, 8000)
      DMMY     =  0.0
      INFL     =  1
C          ARG-R  ARGUMENTS
      DYB    =   YB ( IRB)
      DZB    =   ZB(IRB)
      DA     =   AO
      DAR        =    ARB ( IRB )
      DELEPS   =    1.0
      C      =   CGS
      S    =   -SGS
      DY      =  YS
      DZ    =  ZS
      ITYPE  =  1
```

```
      IF2      =    0
      K        =    1
      ASSIGN   100  TO IRET1
      GO TO 2000
  100 SY   =   1.0
      SZ   =   1.0
      SG       =    SGS
      ASSIGN   200  TO IRET1
      GO TO 5000
  200 CONTINUE
C              CHECK SYMMETRY FLAG-- BRANCH IF EQUAL TO ZERO
      IF ( ND  .EQ. 0 )  GO TO 700
C              PORTION FOR SYMMETRIC CALCULATIONS
      DELEPS   =    ND
      C   =   CGS
      S   =   SSS
      DY    =   -YS
      DZ     =    ZS
      ITYPE    =  1
      IF2      =    0
      K        =    2
C       IF SENDING ELEMENT LIES ON THE Z AXIS.. DON'T USE SYMMETRY
      IF  (  DY  .EQ.  0.0 )   GO TO 700
      ASSIGN  300  TO  IRET1
      GO  TO  2000
  300 CONTINUE
      SY   =  -1.0
      SZ   =  1.0
      SG       =     -SGS
      ASSIGN  400  TO  IRET1
      GO TO 5000
  400 CONTINUE
C              CHECK GROUND EFFECTS FLAG--SKIP IF ZERO
      IF  (  NE  .EQ.  0 )  GO TO 7000
C              PORTION FOR COMBINATION OF SYMMETRY AND GROUND EFFECTS
      ITYPE   =  1
      IF2      =    0
      K        =    3
      DELEPS   =    ND * NE
      C        =    CGS
      S        =    -SGS
      DY     =   -YS
      DZ     =   -ZS
C     IF THE SENDING ELEMENT LIES ON THE Z OR Y AXIS DON'T USE GRND.
C                                   OR SYMMETRY
      IF  (  DZ  .EQ.  0.0)  GO TO 7000
      ASSIGN  500  TO IRET1
      GO  TO  2000
  500 CONTINUE
      SY   =   -1.0
      SG       =     SGS
      SZ   =   -1.0
      ASSIGN  600  TO IRET1
      GO  TO  5000
  600 CONTINUE
      GO TO 800
C               SKIP GROUND EFFECTS CALCULATIONS IF FLAG IS ZERO
  700 IF  (  NE  .EQ.  0 )  GO TO 7000
C               PORTION FOR GROUND EFFECTS ONLY
  800 CONTINUE
      DELEPS   =    NE
      DY   =   YS
      DZ    =    -ZS
C      IF SENDING ELEMENT LIES ON THE Y AXIS DON'T USE GROUND EFFECTS
      IF  (  DZ  .EQ.  0.0) GO TO 7000
      C        =    CGS
      S        =    SGS
      ITYPE    =  1
      IF2      =    0
      K        =    4
      ASSIGN  900  TO IRET1
      GO TO 2000
  900 CONTINUE
      SY   =   1.0
      SZ    =   -1.0
      SG        =     -SGS
      ASSIGN  1000 TO IRET1
      GO TO 5000
 1000 CONTINUE
 1100 RETURN
C
C               CALCULATION OF EFFECTIVE FORCES
C
 2000 CONTINUE
      IF  (IBFS . NE . 0)  GO TO  2010
      CALL  FMZY  ( DYB,        DZB,        DA ,        DAR,        DY,
     *    DZ,         KR,         MACH,       CBAR,
     *    DFZZR,      DFZZI,      DFZYR,      DFZYI       DFYZR,      DFYZI,
     *    DFYYR,      DFYYI,      DMZZR,      DMZZI,      DMZYR,      DMZYI,
     *    DMYZR,      DMYZI,      DMYYR,      DMYYI,      IF1,        IF2  )
      IF (  IPRNT  .NE. 0 )  WRITE ( NPOT, 8010 )  DYB, DZB,  DA,
     *   DAR, DY, DZ,  KR,  MACH,  CBAR,   DFZZR,  DFZZI,   DFZYR,
     *    DFZYI,  DFYZR,   DFYZI,   DFYYR,   DFYYI,   DMZZR,   DMZZI,
     *     DMZYR,   DMZYI,   DMYZR,   DMYZI,   DMYYR,   DMYYI,
     *    IF1,   IF2
      GO TO  2050
 2010 CONTINUE
C
      CALL FZY2   ( XS        , XBLE     , XBTE     , DY       , DZ       ,
     *    DYB      , DZB      , DA       , BETA2    , CBAR     , KR       ,
     *    DFZZR    , DFZZI    , DFZYR    , DFZYI    , DFYZR    , DFYZI    ,
     *    DFYYR    , DFYYI    , IPRNT                                     )
      IF ( IPRNT .NE. 0 )  WRITE ( NPOT , 8010 )
     *   XS,  XBLE,  XBTE,  DY,  DZ,  DYB,  DZB,  DA,
     *   BETA2, DFZZR, DFZZI, DFZYR, DFZYI, DFYYR, DFYYI
C
 2050 CONTINUE
      FWZR      =      C * DFZZR    +  S * DFZYR
      FWZI      =      C * DFZZI    +  S * DFZYI
      FWZ       =      FWZ  +  DELEPS * CMPLX ( FWZR,  FWZI )
      FWYR      =      C * DFYZR    +  S * DFYYR
      FWYI      =      C * DFYZI    +  S * DFYYI
      FWY       =      FWY  +  DELEPS * CMPLX ( FWYR,  FWYI )
      IF  (IBFS . NE . 0)  GO TO  2100
      MWZR      =      C * DMZZR    +  S * DMZYR
      MWZI      =      C * DMZZI    +  S * DMZYI
      MWZ       =      MWZ  +  DELEPS * CMPLX ( MWZR,  MWZI )
      MWYR      =      C * DMYZR    +  S * DMYYR
```

```
      MWYI     =    C * DMYZI  +  S * DMYYI
      MWY      =    MWY  +  DELEPS * CMPLX ( MWYR,  MWYI )
 2100 CONTINUE
      GO TO ( 3000, 6000) , ITYPE
 3000 GO TO IRETI, ( 100,  200,  300,  400,  500,  600,  800,  900,
     *               1000 )
C
C               CALCULATION LOOP FOR ASSOCIATED BODIES
C
 5000 IF (  NAS   .LE. 0 ) GO TO 3000
      I  =  1
      ITYPE    =  2
 5100 IB  =   NASB(I)
C               CHECK TO SEE IF THE ASSOCIATED BODY IS THE RECEIVING
C               BODY.
      IF2       =    0
      IF (  IB   .NE.  IRB )  GO TO 5800
C               IF IT IS DETERMINE IF THE SENDING POINT IS OUTSIDE
C               OR INSIDE THE BODY.
      GO  TO  (  5600,  5500,  5400,  5300 ) , K
 5300 IF (  DYB  .NE. 0.0 )  GO TO 5800
      IF2       =    1
      GO TO 5800
 5400 IF ( DYB .NE. 0.0 )  GO TO 5800
 5500 IF (  DZB  .NE.  0.0   )  GO TO 5800
 5600 IF2       =    1
 5800 CONTINUE
      ETA       =     SY + YS
      ZETA      =     SZ + ZS
      ZBI       =     SZ  +  ZB(IB)
      YBI       =     SY  +  YB(IB)
      DARIB     =     ARB ( IB )
      DAIB      =     AVR ( IB )
      CALL  SUBI  ( DAIB     , ZBI      , YBI      , DARIB    ,
     *              ETA    ,  ZETA   ,  CGS    ,  SG     ,
     *              DMMY   ,  DMMY   ,  DMMY   ,  DY     ,  DZ     ,
     *              DMMY   ,  DMMY   ,  DMMY   ,  DMMY   ,  S      ,
     *              C      ,  INFL   ,  IOUTFL   )
      IF ( IPRNT  .NE.  0 )      WRITE ( NPOT , 8020 )
     *  DAIB,  ZBI,  YBI,  DARIB,  ETA,  ZETA,  CGS,  SG,  DY,
     *          DZ,    S,    C,    INFL,    IOUTFL
      IF ( IOUTFL ) 2000, 3000, 2000
 6000 I = I + 1
      IF ( I - NAS ) 5100,  5100,  3000
 7000 RETURN
 8000 FORMAT ( 14HOENTERING FWMW )
 8010 FORMAT (  5(  5E20.8/) / 2I1)  )
 8020 FORMAT (   7H SUBI../  5E20.8/  5E20.8/  2E20.8,2I20      )
      END


      SUBROUTINE IOF1 (EE,E2,    ETA01,ZET01,ARE,AIM,BRE,BIM,CRE,CIM,
     1                  RISQX,XIIJR,XIIJI)
C    ***   INTEGRATES THE PLANAR PARTS OF THE INCREMENTAL
C          OSCILLATORY KERNELS FOR UNSTEADY CASES
   10 FORMAT  (1H0,    6E20.6/)
      PI    = 3.1415926
      PARN = ETA01**2 - ZET01**2
      FACR = PARN*ARE + ETA01*BRE + CRE
      FACI = PARN*AIM + ETA01*BIM + CIM
      PARNR= BRE/2.0   + ETA01*ARE
      PARNI= BIM/2.0   + ETA01*AIM
      UP    = (ETA01-EE)**2 + ZET01**2
      DOWN  = (ETA01+EE)**2 + ZET01**2
      ARG2 = UP/DOWN
      ALARG2 = ALOG(ARG2)
      TRM2R= PARNR * ALOG(ARG2)
      TRM2I= PARNI * ALOG(ARG2)
      TRM3R= 2.0*EE* ARE
      TRM3I= 2.0*EE* AIM
      AZET = ABS(ZET01)
      IF   ((AZET/EE) . LE . 0.001)  GO TO  100
      TESTO= ABS((RISQX-E2)/(2.0*EE*AZET))
      IF (TESTO.LE.0.0001)  GO TO 110
      COEF = (2.0*EE)/(RISQX-E2)
      ARGA = COEF*ZET01
      TEST = ABS(ARGA)
      IF (TEST.LE.0.3)   GO TO 120
      ARGT = COEF*AZET
      ATANV= ATAN(ARGT)
      FUNCT= ATANA/AZET
      GO TO 170
  100 CONTINUE
      FUNCT= (2.0*EE)/(ETA01**2-E2)
      GO TO 170
  110 CONTINUE
      FUNCT= 0.0
      GO TO 170
  120 CONTINUE
      S       = ARGA**2
      SER   = 1./3.+S*(-1./5.+S*(1./7.+S*(-1./9.+S*(1./11.-S/13.))))
      ALPHA= E2*(COEF**2)*SER
      FUNCT= COEF*(1.0-ALPHA*(ZET01**2)/E2)

  170 CONTINUE
      TRM1R= FACR * FUNCT
      TRM1I= FACI * FUNCT
      XIIJR= TRM1R + TRM2R + TRM3R
      XIIJI= TRM1I + TRM2I + TRM3I
      RETURN
      END


      SUBROUTINE IOF2(EE,E2,    ETA01,ZET01,A2R,A2I,B2R,B2I,C2R,C2I,
     1                  RISQX,DIIJR,DIIJI)
C    ***   INTEGRATES THE NONPLANAR PARTS OF THE INCREMENTAL
C          OSCILLATORY KERNELS FOR UNSTEADY CASES
   10 FORMAT  (1H0,    6E20.6/)
      EPS   = 0.0001
      AZET  = ABS(ZET01)
      DENO  = RISQX-E2
      PARN  = ETA01**2 + ZET01**2
      FACR  = PARN*A2R + ETA01*B2R + C2R
      FACI  = PARN*A2I + ETA01*B2I + C2I
```

```
      ETA02=ETA01**2                                                    IDF20100
      ZETO2= ZETO1**2                                                   IDF20110
      IF  ((AZET/EE) . LE . 0.001)  GO TO  120                          IDF20120
      TESTO= ABS((RISQX-E2)/(2.0*EE*AZET))                              IDF20130
      IF (TESTO.GT.0.1)  GO TO 120                                      IDF20140
      DEN2 = (ETA01+EE)**2+ZETO2                                        IDF20150
      DEN3 = (ETA01-EE)**2+ZETO2                                        IDF20160
      FAC2A= RISQX*ETA01+(ETA02-ZETO2)*EE                               IDF20170
      FAC3A= RISQX*ET 01-(ETA02-ZETO2)*EE                               IDF20180
      FAC2B= RISQX+ETA01*EE                                             IDF20190
      FAC3B= RISQX-ETA01*EE                                             IDF20200
      TRM2R= (FAC2A*A2R+FAC2B*B2R+(ETA01+EE)*C2R)/DEN2                  IDF20210
      TRM2I= (FAC2A*A2I+FAC2B*B2I+(ETA01+EE)*C2I)/DEN2                  IDF20220
      TRM3R=-(FAC3A*A2R+FAC3B*B2R+(ETA01-EE)*C2R)/DEN3                  IDF20230
      TRM3I=-(FAC3A*A2I+FAC3B*B2I+(ETA01-EE)*C2I)/DEN3                  IDF20240
      IF (TESTO.LE.0.0001)  GO TO 110                                   IDF20250
      COEF = (2.0*EE)/(RISQX-E2)                                        IDF20260
      ARGA = COEF*ZETO1                                                 IDF20270
      TEST = ABS(ARGA)                                                  IDF20280
      IF  (TEST.GT.0.3)  GO TO 90                                       IDF20290
      S    = ARGA**2                                                    IDF20300
      SER  = 1./3.+S*(-1./5.+S*(1./7.+S*(-1./9.+S*(1./11.-S/13.))))     IDF20310
      ALPHA= E2*(COEF**2)*SER                                           IDF20320
      FUNCT= COEF*(1.0-ALPHA*(ZETO1**2)/E2)                             IDF20330
      GO TO 100                                                         IDF20340
   90 CONTINUE                                                          IDF20350
      ARGT = COEF*AZET                                                  IDF20360
      ATANA= ATAN(ARGT)                                                 IDF20370
      FUNCT= ATANA/AZET                                                 IDF20380
  100 CONTINUE                                                          IDF20390
      TRM1R= FACR*FUNCT                                                 IDF20400
      TRM1I= FACI*FUNCT                                                 IDF20410
      DIIJR= (TRM1R + TRM2R + TRM3R)/(2.0*ZETO2)                        IDF20420
      DIIJI= (TRM1I + TRM2I + TRM3I)/(2.0*ZETO2)                        IDF20430
      GO TO 170                                                         IDF20440
  110 CONTINUE                                                          IDF20450
      FUNCT= 0.0                                                        IDF20460
      GO TO 100                                                         IDF20470
  120 CONTINUE                                                          IDF20480
      DENA = (ETA01+EE)**2 + ZETO1**2                                   IDF20490
      DENB = (ETA01-EE)**2 + ZETO1**2                                   IDF20500
      UP1R = 2.0*(E2*A2R + C2R)                                         IDF20510
      UP1I = 2.0*(E2*A2I + C2I)                                         IDF20520
      UP2R = 4.0*E2*ETA01*B2R                                           IDF20530
      UP2I = 4.0*E2*ETA01*B2I                                           IDF20540
      TRM1R= (UP1R *(RISQX+E2) + UP2R )/(DENA*DENB)                     IDF20550
      TRM1I= (UP1I *(RISQX+E2) + UP2I )/(DENA*DENB)                     IDF20560
      IF  ((AZET/EE) . LE . 0.001)  GO TO  130                          IDF20570
      COEF = (2.0*EE)/(RISQX-E2)                                        IDF20580
      ARGA = COEF*ZETO1                                                 IDF20590
      TEST = ABS(ARGA)                                                  IDF20600
      IF  (TEST.GT.0.3)  GO TO 125                                      IDF20610
      S    = ARGA**2                                                    IDF20620
      SER  = 1./3.+S*(-1./5.+S*(1./7.+S*(-1./9.+S*(1./11.-S/13.))))     IDF20630
      ALPHA= E2*(COEF**2)*SER                                           IDF20640
      FUNCT= COEF*(1.0-ALPHA*(ZETO1**2)/E2)                             IDF20650
      GO TO 140                                                         IDF20660
  125 CONTINUE                                                          IDF20670
      ARGT= COEF*AZET                                                   IDF20680
      ATANA= ATAN(ARGT)                                                 IDF20690
      FUNCT= ATANA/AZET                                                 IDF20700
      ALPHA= (E2/ZETO2)*(1.0-FUNCT*(DENO/(2.0*EE)))                     IDF20710
      GO TO 140                                                         IDF20720
  130 CONTINUE                                                          IDF20730
      ALPHA= ((2.0*E2)/(ETA02-E2))**2                                   IDF20740
  140 CONTINUE                                                          IDF20750
      TRM2R= -ALPHA*FACR/E2                                             IDF20760
      TRM2I= -ALPHA*FACI/E2                                             IDF20770
      DIIJR= EE*(TRM1R + TRM2R)/DENO                                    IDF20780
      DIIJI= EE*(TRM1I + TRM2I)/DENO                                    IDF20790
  170 CONTINUE                                                          IDF20800
      RETURN                                                            IDF20810
      END                                                               IDF20820
```

```
      SUBROUTINE INCR0(AX,AY,AZ,AX1,AY1,AZ1,AX2,AY2,AZ2,SGR,CGR,SGS,CGS,INCR0010
     1          K ,FL,BETA,SDELX,DELY,DELR,DELI,IO,IR,NBXS,NCPNB,LHS,   INCR0020
     2          NOBLE,IMG,NOBI,IMGS,  USE1 ,      USE2,       USE3,     INCR0030
     3          U E4,       XUSE1,     XUSE2,      XUSE3,      XUSE4 )  INCR0035
C   ***   CALCULATES THE UNSTEADY PART OF THE INFLUENCE COEFFICIENT     INCR0040
C         MATRIX ELEMENTS USING  SUBROUTINES  KERNEL, IDF1  AND  IDF2   INCR0045
      REAL   K10,K20,K1RT1,K1IT1,K2RT2P,K2IT2P,K1OT1,K2OT2P, KR,M       INCR0050
      COMMON  /DLM/  K10,K20,K1RT1,K1IT1,K2RT2P,K2IT2P,K1OT1,K2OT2P,F2  INCR0060
      COMMON     /KDS/ IND,KD1R,KD1I,KD2R,KD2I                          INCR0070
      REAL        KD1R,KD1I, KD2R,KD2I                                  INCR0080
      DIMENSION   USE1(50,14), USE2(50,14), USE3(50,14), USE4(50,14)    INCR0090
      DIMENSION  XUSE1(50,14),XUSE2(50,14),XUSE3(50,14),XUSE4(50,14)    INCR0100
    6 FORMAT (1H0,15I5)                                                 INCR0110
   10 FORMAT (1H0,6E20.9)                                               INCR0120
C***********************************************************************INCR0130
C                                                                       INCR0140
C  *****   THE FOLLOWING LIST IS THE NAMES OF THE KERNEL VALUES         INCR0150
C  *****   USED IN THE SAVING PROCESS OF KERNELS                        INCR0160
C                                                                       INCR0170
C  **  DKRO = REAL PART OF THE PLANAR KERNEL  *  OUTBOARD POINT         INCR0180
C  **  DKIO = IMAGINARY PART OF THE PLANAR KERNEL  *  OUTBOARD POINT    INCR0190
C  **  XKRO = REAL PART OF THE NONPLANAR KERNEL  *  OUTBOARD POINT      INCR0200
C  **  XKIO = IMAGINARY PART OF THE NONPLANAR KERNEL  *  OUTBOARD POINT INCR0210
C  **  DKRI = REAL PART OF THE PLANAR KERNEL .*   INBOARD POINT         INCR0220
C  **  DKII = IMAGINARY PART OF THE PLANAR KERNEL  *   INBOARD POINT    INCR0230
C  **  XKRI = REAL PART OF THE NONPLANAR KERNEL  *   INBOARD POINT      INCR0240
C  **  XKII = IMAGINARY PART OF THE NONPLANAR KERNEL  *   INBOARD POINT INCR0250
C                                                                       INCR0260
C  *****   THE FOLLOWING TABLE EXPLAINS THE USAGE OF THE                INCR0270
C  *****   STORED KERNEL ARRAYS                                         INCR0280
C                                                                       INCR0290
C***********************************************************************INCR0300
C                                                                       INCR0310
C  SENDING POINT IN   *  SENDING POINT IN   *  VALUE OF INDEX 'J' WHEN  INCR0320
C  RIGHT HALF-PLANE   *  LEFT  HALF-PLANE   *       SENDING POINT IS    INCR0330
C     (Y.GE.0)        *     (Y.LT.0)        *...........................INCR0340
C.....................*.....................* ON LIFTING . INSIDE ASSOC.INCR0350
C                     *                     *    SURFACE  .  BODY (NOBI) INCR0360
C'OUTBOARD'.  SAVED   * 'INBOARD'.  SAVED   *...........................INCR0370
C  KERNELS .    IN    *  KERNELS .    IN    *UPPER* LOWER. UPPER * LOWERINCR0380
C          .          *            .        * HALF-PLANE . HALF-PLANE   INCR0390
```

```
C             .         *         .         *    .    .      .       INCR0400
C*********************************************************************INCR0410
C             .         *         .         *    .    .      .       INCR0420
C   DKRO     . USE1(I,J)*  DKRI    . USE3(I,J)*    .    .      .       INCR0430
C   DKIO     . USE2(I,J)*  DKII    . USE4(I,J)*    .    .      .       INCR0440
C             .         *         .         * 1  .  2 .2+NOBI .8+NOBIINCR0450
C   XKRO     .XUSE1(I,J)*  XKRI    .XUSE3(I,J)*    .    .      .       INCR0460
C   XKIO     .XUSE2(I,J)*  XKII    .XUSE4(I,J)*    .    .      .       INCR0470
C             .         *         .         *    .    .      .       INCR0480
C*********************************************************************INCR0490
C                                                                     INCR0500
      IND  = 1                                                        INCR0510
      M = SQRT(1.0 - BETA**2)                                         INCR0520
      BR   = FL/2.                                                    INCR0530
      EPS = 0.00001                                                   INCR0540
      PI   = 3.14159265                                               INCR0550
      YDELX = SDELX                                                   INCR0560
      XDELY =  DELY                                                   INCR0570
      EE = 0.5*XDELY                                                  INCR0580
      E2   = EE**2                                                    INCR0590
      DELR = 0.0                                                      INCR0600
      DELI = 0.0                                                      INCR0610
      T1S  = 0.0                                                      INCR0620
      T2S  = 0.0                                                      INCR0630
      AT1S = 0.0                                                      INCR0640
      AT2S = 0.0                                                      INCR0650
      T1   = 0.0                                                      INCR0660
      T2   = 0.0                                                      INCR0670
      COUNT =0.                                                       INCR0680
      XO=AX                                                           INCR0690
      YO=AY                                                           INCR0700
      ZO=AZ                                                           INCR0710
   80 CONTINUE                                                        INCR0720
      CALL        TKER  (XO,YO,ZO,KR,BR,SGR,CGR,SGS,CGS,T1,T2,M)      INCR0730
      AT1  = ABS(T1)                                                  INCR0740
      AT2  = ABS(T2)                                                  INCR0750
      IF (AT1.GT.AT1S)  AT1S=AT1                                      INCR0760
      IF (AT2.GT.AT2S)  AT2S=AT2                                      INCR0770
      IF (COUNT)130,90,150                                            INCR0780
   90 DKRC  = K1RT1 - K1OT1                                           INCR0790
      DKIC  = K1IT1                                                   INCR0800
      XKRC  = K2RT2P-K2OT2P                                           INCR0810
      XKIC  = K2IT2P                                                  INCR0820
      AT2 = ABS(T2)                                                   INCR0830
      JO = 1                                                          INCR0840
      IF  (IMG.NE.0)  JO = 2 + NOBI                                   INCR0850
      IF (NOBLE.NE.0) JO = 2                                          INCR0860
      IF  (IMG.NE.0.AND.NOBLE.NE.0)  JO = 8 + NOBI                    INCR0870
      IF(IMG.EQ.1 . AND . IMGS.EQ.0)  GO TO  110                      INCR0875
      IF (IR.LE.NCPNB) GO TO  110                                     INCR0880
      IF (IR.GT.NBXS) GO TO 110                                       INCR0890
      IF (LHS.NE.0) GO TO  100                                        INCR0900
      DKRI = USE1(IO,JO)                                              INCR0910
      DKII = USE2(IO,JO)                                              INCR0920
      XKRI  = XUSE1(IO,JO)                                            INCR0930
      XKII  = XUSE2(IO,JO)                                            INCR0940
      XO    = AX2                                                     INCR0950
      YO    = AY2                                                     INCR0960
      ZO = AZ2                                                        INCR0970
      CALL      TKER  (XO,YO,ZO,KR,BR,SGR,CGR,SGS,CGS,T1,T2,M)        INCR0980
      AT1  = ABS(T1)                                                  INCR0990
      AT2  = ABS(T2)                                                  INCR1000
      IF (AT1.GT.AT1S)  AT1S=AT1                                      INCR1010
      IF (AT2.GT.AT2S)  AT2S=AT2                                      INCR1020
      DKRO  = K1RT1 - K1OT1                                           INCR1030
      DKIO  = K1IT1                                                   INCR1040
      XKRO  = K2RT2P-K2OT2P                                           INCR1050
      XKIO  = K2IT2P                                                  INCR1060
      GO TO  170                                                      INCR1070
  100 XO = AX1                                                        INCR1080
      YO = AY1                                                        INCR1090
      ZO = AZ1                                                        INCR1100
      CALL      TKER  (XO,YO,ZO,KR,BR,SGR,CGR,SGS,CGS,T1,T2,M)        INCR1110
      AT1  = ABS(T1)                                                  INCR1120
      AT2  = ABS(T2)                                                  INCR1130
      IF (AT1.GT.AT1S)  AT1S=AT1                                      INCR1140
      IF (AT2.GT.AT2S)  AT2S=AT2                                      INCR1150
      DKRI  = K1RT1 - K1OT1                                           INCR1160
      DKII  = K1IT1                                                   INCR1170
      XKRI  = K2RT2P-K2OT2P                                           INCR1180
      XKII  = K2IT2P                                                  INCR1190
      DKRO = USE3(IO,JO)                                              INCR1200
      DKIO = USE4(IO,JO)                                              INCR1210
      XKRO  = XUSE3(IO,JO)                                            INCR1220
      XKIO  = XUSE4(IO,JO)                                            INCR1230
      GO TO  160                                                      INCR1240
  110 CONTINUE                                                        INCR1250
      COUNT =-1.                                                      INCR1260
  120 XO = AX1                                                        INCR1270
      YO = AY1                                                        INCR1280
      ZO = AZ1                                                        INCR1290
      GO TO 80                                                        INCR1300
  130 DKRI  = K1RT1 - K1OT1                                           INCR1310
      DKII  = K1IT1                                                   INCR1320
      XKRI  = K2RT2P-K2OT2P                                           INCR1330
      XKII  = K2IT2P                                                  INCR1340
  140 COUNT = 1.                                                      INCR1350
      XO=AX2                                                          INCR1360
      YO=AY2                                                          INCR1370
      ZO=AZ2                                                          INCR1380
      GO TO 80                                                        INCR1390
  150 DKRO  = K1RT1 - K1OT1                                           INCR1400
      DKIO  = K1IT1                                                   INCR1410
      XKRO  = K2RT2P-K2OT2P                                           INCR1420
      XKIO  = K2IT2P                                                  INCR1430
      JO = 1                                                          INCR1440
      IF  (IMG.NE.0)  JO = 2 + NOBI                                   INCR1450
      IF (NOBLE.NE.0) JO = 2                                          INCR1460
      IF  (IMG.NE.0.AND.NOBLE.NE.0)  JO = 8 + NOBI                    INCR1470
      IF (LHS.EQ.0) GO TO  170                                        INCR1480
  160 USE3(IO,JO) = DKRI                                              INCR1490
      USE4(IO,JO) = DKII                                              INCR1500
      XUSE3(IO,JO)= XKRI                                              INCR1510
      XUSE4(IO,JO)= XKII                                              INCR1520
      GO TO 180                                                       INCR1530
  170 USE1(IO,JO) = DKRO                                              INCR1540
      USE2(IO,JO) = DKIO                                              INCR1550
      XUSE1(IO,JO)= XKRO                                              INCR1560
```

```
      XUSE2(IO,JO)= XKIO                                                INCR1570
  180 CONTINUE                                                          INCR1580
      XO = AX                                                           INCR1590
      YO = AY                                                           INCR1600
      ZO = AZ                                                           INCR1610
      ZERO = 0.0                                                        INCR1620
      XIIJR= 0.                                                         INCR1630
      XIIJI= 0.                                                         INCR1640
      DIIJR= 0.0                                                        INCR1650
      DIIJI= 0.0                                                        INCR1660
      PI    = 3.1415926                                                 INCR1670
      XMULT =((XDELX)/(8.0*PI))                                         INCR1680
      IF ((YO.EQ.ZERO).AND.(ZO.EQ.ZERO))  GO TO 220                     INCR1690
      IF   ((ZO.EQ.ZERO).AND.(SGS.EQ.ZERO))  GO TO 230                  INCR1700
  190 ETA01 = YO*CGS + ZO*SGS                                           INCR1710
      ZETO1 =-YO*SGS + ZO*CGS                                           INCR1720
      AZETO = ABS(ZETO1)                                                INCR1730
      IF (AZETO.LE.0.0001)  ZETO1 = 0.                                  INCR1740
  200 RISQX = ETA01**2 + ZETO1**2                                       INCR1750
  210 ARE = (DKRI -2.*DKRC + DKRO)/(2.0*E2)                              INCR1760
      AIM = (DKII -2.*DKIC + DKIO)/(2.0*E2)                              INCR1770
      BRE = (DKRO - DKRI)/(2.0*EE)                                      INCR1780
      BIM = (DKIO - DKII)/(2.0*EE)                                      INCR1790
      CRE =  DKRC                                                       INCR1900
      CIM =  DKIC                                                       INCR1810
      GO TO 250                                                         INCR1820
  220 ETA01 = 0.0                                                       INCR1830
      ZETO1 = 0.0                                                       INCR1840
      RISQX = 0.0                                                       INCR1850
      GO TO  210                                                        INCR1860
  230 ETA01  = YO*CGS                                                   INCR1870
      ZETO1  = 0.                                                       INCR1980
  240 RISQX  = ETA01**2                                                 INCR1893
      GO TO  210                                                        INCR1900
  250 CONTINUE                                                          INCR1910
      IF (AT1S.EQ.0.0)  GO TO 255                                       INCR1920
      CALL        IDF1 (EE,E2,    ETA01,ZETO1,ARE,AIM,BRE,BIM,CRE,CIM,  INCR1930
     1                     RISQX,XIIJR,XIIJI)                           INCR1940
      DELR =XMULT*XIIJR                                                 INCR1950
      DELI =XMULT*XIIJI                                                 INCR1960
  255 CONTINUE                                                          INCR1970
      IF (AT2S.EQ.0.0)  GO TO 260                                       INCR1980
      A2R  = (XKRI - 2.0*XKRC + XKRO)/(2.0*E2)                          INCR1990
      A2I  = (XKII - 2.0*XKIC + XKIO)/(2.0*E2)                          INCR2000
      B2R  = (XKRO - XKRI)/(2.0*EE)                                     INCR2010
      B2I  = (XKIO - XKII)/(2.0*EE)                                     INCR2020
      C2R  =  XKRC                                                      INCR2030
      C2I  =  XKIC                                                      INCR2040
      CALL        IDF2(EE,E2,    ETA01,ZETO1,A2R,A2I,B2R,B2I,C2R,C2I,   INCR2050
     1                     RISQX,DIIJR,DIIJI)                           INCR2060
      DELR = DELR + XMULT*DIIJR                                         INCR2070
      DELI = DELI + XMULT*DIIJI                                         INCR2080
  260 CONTINUE                                                          INCR2082
C    ***                                                                INCR2084
C    ***                                                                INCR2086
  265 RETURN                                                            INCR2090
      END                                                               INCR2100
```

```
CREADO         SUBROUTINE TO READ DZ OR DY FROM TAPE                    READ0010
C              USE OF AN ARRAY TYPE OF INPUT TO INCREASE SPEED          READ0020
      SUBROUTINE  READD ( D, N, NTAPE)                                  READ0030
      COMPLEX  D(N)                                                     READ0040
      READ ( NTAPE ) (ROW, IT, D                                        READ0050
      RETURN                                                            READ0060
      END                                                               READ0070
```

```
CRWREC                                                                  RWRE0010
      SUBROUTINE RWREC( IFLAG, NTAP , A       , NCWORD, NUMBR)          RWRE0010
C   ***    READS OR WRITES AN UNFORMATTED RECORD ON TAPE ACCORDING TO   RWRE0020
C          EITHER OF THE FOLLOWING FOUR OPTIONS  --                     RWRE0030
C          IFLAG = 0  **  WRITE ARRAY   A                               RWRE0040
C          IFLAG = 1  **  READ  ARRAY   A                               RWRE0050
C          IFLAG = 2  **  WRITE ARRAY   A  PRECEDED BY   NUMBR          RWRE0060
C          IFLAG = 3  **  READ  ARRAY   A  PRECEDED BY   NUMBR          RWRE0070
C   ***    IN EACH CASE THE LENGTH OF THE RECORD  (DIMENSION OF         RWRE0080
C          THE ARRAY)  IS   NCWORD                                      RWRE0090
      COMPLEX     A(NCWORD)                                             RWRE0100
      IFLP1  = IFLAG + 1                                                RWRE0110
      GO TO  (10, 20, 30, 40),  IFLP1                                   RWRE0120
   10 WRITE   (NTAP)            A                                       RWRE0130
      RETURN                                                            RWRE0140
   20 READ    (NTAP)            A                                       RWRE0150
      RETURN                                                            RWRE0160
   30 WRITE   (NTAP)  NUMBR,    A                                       RWRE0170
      RETURN                                                            RWRE0180
   40 READ    (NTAP)  NUMBR,    A                                       RWRE0190
      RETURN                                                            RWRE0200
      END                                                               RWRE0210
```

```
      SUBROUTINE SNPDF(SL,CL,TL,SGS,CGS,SGR,CGR,XO,YO,ZO,EE,DIJ,BETA,CV)SNPD0010
C   ***   CALCULATES THE STEADY PART OF THE INFLUENCE COEFFICIENT       SNPD0012
C            MATRIX ELEMENTS                                            SNPD0014
      REAL    KR,KROBR                                                  SNPD0020
   10 FORMAT  (1HO, 6F15.8)

      TEST1= 0.9999
      TEST2= 0.0001*EE                                                  SNPD0060
C  ***  TEST1  AND TEST2  SERVE AS A MEASURE OF  'NEARNESS'  WITH       SNPD0070
C  RESPECT TO THE BOUND-  AND TRAILING VORTICES RESPECTIVELY  --  SEE   SNPD0080
C  TESTS BELOW                                                          SNPD0090
C  ***  NOTE  THAT THE MACH NUMBER EFFECT IS ACCOUNTED FOR BY STRETCHINGSNPD0100
C  THE  X-COORDINATES AND THE  SWEEP ANGLE OF THE BOUND VORTEX LINE     SNPD0110
      TLB  = TL/BETA                                                    SNPD0120
      SQTLB= SQRT(1.0+TLB**2)                                           SNPD0130
      SLB  = TLB/SQTLB                                                  SNPD0140
      CLB  = 1.0/SQTLB                                                  SNPD0150
      CAVE = CV                                                         SNPD0160
      CLSGS = CLB*SGS                                                   SNPD0170
      CLCGS = CLB*CGS                                                   SNPD0180
      EX =EE*TLB                                                        SNPD0190
```

```
      EY =EE*CGS                                                          SNPD0200
      EZ =EE*SGS                                                          SNPD0210
      XOB =  XO / BETA                                                    SNPD0220
      RIX = XOB+ EX                                                       SNPD0230
      RIY = YO + EY                                                       SNPD0240
      RIZ = ZO + EZ                                                       SNPD0250
      RIMAG = SQRT(RIX**2 + RIY**2 + RIZ**2)                              SNPD0260
      ROX = XOB- EX                                                       SNPD0270
      ROY = YO - EY                                                       SNPD0280
      ROZ = ZO - EZ                                                       SNPD0290
      ROMAG = SQRT(ROX**2 + ROY**2 + ROZ**2)                              SNPD0300
      CAB = (RIX*SLB+ RIY*CLCGS + RIZ*CLSGS)/RIMAG                        SNPD0310
      CBB = (ROX*SLB+ ROY*CLCGS + ROZ*CLSGS)/ROMAG                        SNPD0320
      CBI = -RIX/RIMAG                                                    SNPD0330
      CAO =  ROX/ROMAG                                                    SNPD0340
      RICAB = RIMAG*CAB                                                   SNPD0350
      DBX = RIX - RICAB*SLB                                               SNPD0360
      DBY = RIY - RICAB*CLCGS                                             SNPD0370
      DBZ = RIZ - RICAB*CLSGS                                             SNPD0380
      DB2 = DBX**2 + DBY**2 + DBZ**2                                      SNPD0390
      DI2 = RIY**2 + RIZ**2                                               SNPD0400
      DO2 = ROY**2 + ROZ**2                                               SNPD0410
      ACAB = ABS(CAB)                                                     SNPD0420
      ACBB = ABS(CBB)                                                     SNPD0430
C  ***  THE FOLLOWING IS A TEST TO SEE IF THE RECEIVING POINT LIES ON     SNPD0440
C  OR NEAR THE BOUND VORTEX  --  IF SO, THE CONTRIBUTION OF THE BOUND     SNPD0450
C  VORTEX IS SET TO ZERO                                                  SNPD0460
      IF (ACAB.GT.TEST1)  GO TO  30                                       SNPD0470
      IF (ACBB.GT.TEST1)  GO TO  30                                       SNPD0480
      CACB =(CAB - CBB)/DB2                                               SNPD0490
      GO TO  50                                                           SNPD0500
   30 IF (CAB*CBB)40,50,50                                                SNPD0510
   40 CACB = 0.                                                           SNPD0520
      GO TO 60                                                            SNPD0530
   50 CACB = 0.5*ABS((1./RIMAG**2)-(1./ROMAG**2))                         SNPD0540
   60 CONTINUE                                                            SNPD0550
      VBY = CACB * (DBX*CLSGS - DBZ*SLB)                                  SNPD0560
      VBZ = CACB * (DBY*SLB- DBX*CLCGS)                                   SNPD0570
C  ***  TEST TO SEE IF THE RECEIVING POINT LIES ON OR NEAR THE            SNPD0580
C  INBOARD TRAILING VORTEX  --  IF SO, THE CONTRIBUTION OF THE            SNPD0590
C  INBOARD TRAILING VORTEX IS SET TO ZERO                                 SNPD0600
      IF (DI2.GT.TEST2)  GO TO 62                                         SNPD0610
      VIY  = 0.0                                                          SNPD0620
      VIZ  = 0.0                                                          SNPD0630
      GO TO  64                                                           SNPD0640
   62 CONTINUE                                                            SNPD0650
      ONECBI = (1.0 - CBI)/DI2                                            SNPD0660
      VIY = ONECBI*RIZ                                                    SNPD0670
      VIZ = -ONECBI*RIY                                                   SNPD0680
   64 CONTINUE                                                            SNPD0690
C  ***  TEST TO SEE IF THE RECEIVING POINT LIES ON OR NEAR THE            SNPD0700
C  OUTBOARD TRAILING VORTEX  --  IF SO, THE CONTRIBUTION OF THE           SNPD0710
C  OUTBOARD TRAILING VORTEX IS SET TO ZERO                                SNPD0720
      IF (DO2.GT.TEST2)  GO TO 66                                         SNPD0730
      VOY  = 0.0                                                          SNPD0740
      VOZ  = 0.0                                                          SNPD0750
      GO TO  68                                                           SNPD0760
   66 CONTINUE                                                            SNPD0770
      CAOONE = (1.0 + CAO)/DO2                                            SNPD0780
      VOY = -CAOONE*ROZ                                                   SNPD0790
      VOZ = CAOONE*ROY                                                    SNPD0800
   68 CONTINUE                                                            SNPD0810
      VX = DBZ*CLCGS - DBY*CLSGS                                          SNPD0820
      VY = VBY + VIY + VOY                                                SNPD0830
      VZ = VBZ + VIZ + VOZ                                                SNPD0840
      WW= VY*SGR - VZ*CGR                                                 SNPD0850
      DIJ = WW* CAVE / 25.132741                                          SNPD0860
   70 CONTINUE                                                            SNPD0870
      RETURN                                                              SNPD0880
      END                                                                 SNPD0890


      SUBROUTINE SUBI(DA,DZB,DYB,DAR,DETA,DZETA,DCGAM,DSGAM,DEE,DXI,TL, SUBI0010
     1 DETAI,DZETAI,DCGAMI,DSGAMI,DEEI,DTLAMI,DMUY,DMUZ,INFL,IOUTFL)    SUBI0020
C   ***    COMPUTES THE IMAGE POINT COORDINATES INSIDE ASSOCIATED         SUBI0022
C          BODIES AND THE  MU-Z  MU-Y  ELEMENTS USED IN SUBROUTINE FMM    SUBI0024
   15 FORMAT  (1H0,15I5)                                                  SUBI0030
   35 FORMAT  (1H0,6E20.8)                                                SUBI0040
      EPS  = 0.001                                                        SUBI0050
      DMUY = 0.0                                                          SUBI0060
      DMUZ = 0.0                                                          SUBI0070
      IGO  = 1                                                            SUBI0080
      PSQR = SQRT(((DETA-DYB)*DAR)**2 + (DZETA-DZB)**2)                   SUBI0090
      COSTH= (DETA -DYB)*DAR/PSQR                                         SUBI0100
      SINTH= (DZETA-DZB)/PSQR                                             SUBI0110
      CT2  = COSTH*COSTH                                                  SUBI0120
      ST2  = SINTH*SINTH                                                  SUBI0130
      CT3  = COSTH*CT2                                                    SUBI0140
      ST3  = SINTH*ST2                                                    SUBI0150
      YCBAR= DA*(1.0-DAR*DAR)*CT3 + DYB                                   SUBI0160
      ZCBAR= DA*(DAR*DAR-1.0)*ST3/DAR + DZB                               SUBI0170
      PAREN= ST2 + DAR*DAR*CT2                                            SUBI0180
      PAR3 = PAREN*PAREN**2                                               SUBI0190
      ABAR = DA*SQRT(PAR3)/DAR                                            SUBI0200
      ABAR2= ABAR*ABAR                                                    SUBI0210
      IF (INFL.NE.0)  GO TO  300                                          SUBI0220
      ETA1 = DETA - DEE*DCGAM                                             SUBI0230
      ETA2 = DETA + DEE*DCGAM                                             SUBI0240
      ZETA1= DZETA- DEE*DSGAM                                             SUBI0250
      ZETA2= DZETA+ DEE*DSGAM                                             SUBI0260
      RHO12= (ETA1 - YCBAR)**2 + (ZETA1-ZCBAR)**2                         SUBI0270
      RHO22= (ETA2 - YCBAR)**2 + (ZETA2-ZCBAR)**2                         SUBI0280
      ETAI1= YCBAR + (ETA1-YCBAR)*ABAR2/RHO12                             SUBI0290
      ETAI2= YCBAR + (ETA2-YCBAR)*ABAR2/RHO22                             SUBI0300
      ZETI1= ZCBAR +(ZETA1-ZCBAR)*ABAR2/RHO12                             SUBI0310
      ZETI2= ZCBAR +(ZETA2-ZCBAR)*ABAR2/RHO22                             SUBI0320
      DEEI = SQRT((ETAI2-ETAI1)**2 + (ZETI2-ZETI1)**2) / 2.0              SUBI0330
      DETAI  = (ETAI1 + ETAI2)/2.0                                        SUBI0340
      DZETAI = (ZETI1 + ZETI2)/2.0                                        SUBI0350
      DCGAMI = -(ETAI2 - ETAI1)/(2.0*DEEI)                                SUBI0360
      DSGAMI= -(ZETI2 - ZETI1)/(2.0*DEEI)                                 SUBI0370
      DXI1 = DXI - DEE*TL                                                 SUBI0380
      DXI2 = DXI + DEE*TL                                                 SUBI0390
      DELXI  = DXI1 - DXI2                                                SUBI0400
      DTLAMI = DELXI/(2.0*DEEI)                                           SUBI0410
```

```
      IF ((ABS(DAR-1.0)).LE.0.0001)  GO TO  420                 SUBI0420
      GO TO  301                                                 SUBI0430
  300 CONTINUE                                                   SUBI0440
      RHO2 = (DETA -YCBAR)**2 + (DZETA -ZCBAR)**2                SUBI0450
      RHO4 = RHO2*RHO2                                           SUBI0460
      DETAI  = YCBAR + (DETA -YCBAR)*ABAR2/RHO2                  SUBI0470
      DZETAI = ZCBAR + (DZETA-ZCBAR)*ABAR2/RHO2                  SUBI0480
  301 CONTINUE                                                   SUBI0490
      GO TO  (302,303,304),  IGO                                 SUBI0500
  302 CONTINUE                                                   SUBI0510
      XETAI  = DETAI                                             SUBI0520
      XZETAI = DZETAI                                            SUBI0530
      GO TO  305                                                 SUBI0540
  303 CONTINUE                                                   SUBI0550
      XETAI  = ETAI1                                             SUBI0560
      XZETAI = ZETI1                                             SUBI0570
      GO TO  307                                                 SUBI0580
  304 CONTINUE                                                   SUBI0590
      XETAI  = ETAI2                                             SUBI0600
      XZETAI = ZETI2                                             SUBI0610
      GO TO  307                                                 SUBI0620
  305 CONTINUE                                                   SUBI0630
      BELL = DA*DAR                                              SUBI0640
      AELL = DA*SQRT(ABS(DAR*DAR-1.0))                           SUBI0650
  307 CONTINUE                                                   SUBI0660
      IF (DAR.LT.1.0)  GO TO  310                                SUBI0670
      DYBM = DYB-EPS                                             SUBI0672
      DYBP = DYB+EPS                                             SUBI0674
      IF  (DETA .GE.DYB .AND.XETAI .LT.DYBM) GO TO  325          SUBI0680
      IF  (DETA .LE.DYB .AND.XETAI .GT.DYBP) GO TO  325          SUBI0690
      ELLIPS = 2.0*DA*DAR                                        SUBI0700
      PART1 = (AELL + (XZETAI - DZB))**2                         SUBI0710
      PART2 = (XETAI - DYB)**2                                   SUBI0720
      PART3 = (AELL - (XZETAI - DZB))**2                         SUBI0730
      GO TO  320                                                 SUBI0740
  310 CONTINUE                                                   SUBI0750
      DZBM = DZB-EPS                                             SUBI0752
      DZBP = DZB+EPS                                             SUBI0754
      IF  (DZETA.GE.DZB .AND.XZETAI.LT.DZBM) GO TO  325          SUBI0760
      IF  (DZETA.LE.DZB .AND.XZETAI.GT.DZBP) GO TO  325          SUBI0770
      ELLIPS = 2.0*DA                                            SUBI0780
      PART1 = (AELL + (XETAI  - DYB))**2                         SUBI0790
      PART2 = (XZETAI- DZB)**2                                   SUBI0800
      PART3 = (AELL - (XETAI  - DYB))**2                         SUBI0810
  320 CONTINUE                                                   SUBI0820
      TEST = SQRT(PART1+PART2) + SQRT(PART3+PART2)               SUBI0830
      TEDIF=      TEST-ELLIPS                                    SUBI0840
      IF (INFL.EQ.0)  GO TO  400                                 SUBI0850
      IF (TEDIF.LE.EPS)  GO TO  330                              SUBI0860
  325 CONTINUE                                                   SUBI0870
      IOUTFL = 0                                                 SUBI0880
      GO TO  500                                                 SUBI0890
  330 CONTINUE                                                   SUBI0900
      IOUTFL = 1                                                 SUBI0910
      TRM1 = (DETA -YCBAR)**2 - (DZETA -ZCBAR)**2                SUBI0920
      TRM2 = 2.0 *(DETA-YCBAR)*(DZETA-ZCBAR)                     SUBI0930
      OMUY = -(-DSGAM*TRM1 + DCGAM*TRM2)*ABAR2/RHO4              SUBI0940
      OMUZ = -(-DSGAM*TRM2 - DCGAM*TRM1)*ABAR2/RHO4              SUBI0950
                                                                 SUBI0960
      GO TO 500                                                  SUBI0970
  400 CONTINUE                                                   SUBI0980
      IF (TEDIF.GT.EPS)  GO TO  325                              SUBI0990
      IF  (IGO.EQ.3)  GO TO 420                                  SUBI1000
      IGO  = IGO+1                                               SUBI1010
      GO TO  301                                                 SUBI1020
  420 CONTINUE                                                   SUBI1030
      IOUTFL = 1                                                 SUBI1040
  500 CONTINUE                                                   SUBI1050
      RETURN                                                     SUBI1060
      END                                                        SUBI1070

      SUBROUTINE TIME(N)                                         TIME0010
C                                                                TIME0020
C     N =  0,   CALL TSETQ                                       TIME0030
C                                                                TIME0040
C     N =  1,   PRINT TIME SINCE LAST CALL TO TIME.              TIME0050
C                                                                TIME0060
C     N =  2,   PRINT TIME SINCE LAST CALL TO TSETQ.             TIME0070
C                                                                TIME0080
C     N = -1,   SAME AS N = 1, BUT ALSO CALL TSETQ.              TIME0090
C                                                                TIME0100
C     N = -2,   SAME AS N = 2, BUT ALSO CALL TSETQ.              TIME0110
C                                                                TIME0120
   10 FORMAT ('0*****'/'0EXECUTION TIME(MINUTES)'/' CPU',T18,F8.4/' I/O'TIME0130
     1  ,T18,F8.4/' TOTAL',T18,F8.4)                             TIME0140
      DATA C/.1666666E-1/                                        TIME0150
      IF (N.EQ.0) GO TO 40                                       TIME0160
      CALL TIMEQ (TCPU,TIO)                                      TIME0170
      TCPU = TCPU * C                                            TIME0180
      TIO = TIO * C                                              TIME0190
      T = TCPU + TIO                                             TIME0200
      IF (IABS(N).EQ.1) GO TO 20                                 TIME0210
      TCPUP = TCPU                                               TIME0220
      TIOP = TIO                                                 TIME0230
      TP = T                                                     TIME0240
      GO TO 30                                                   TIME0250
   20 TCPUP = TCPU - T1                                          TIME0260
      TIOP = TIO - T2                                            TIME0270
      TP = T - T3                                                TIME0280
   30 WRITE (6,10) TCPUP, TIOP, TP                               TIME0290
      IF (N.LT.0) GO TO 40                                       TIME0300
      T1 = TCPU                                                  TIME0310
      T2 = TIO                                                   TIME0320
      T3 = T                                                     TIME0330
      RETURN                                                     TIME0340
   40 CALL TSETQ                                                 TIME0350
      T1 = 0.0                                                   TIME0360
      T2 = 0.0                                                   TIME0370
      T3 = 0.0                                                   TIME0380
      RETURN                                                     TIME0390
      END                                                        TIME0400
```

```
      SUBROUTINE TKER  (XO,YO,ZO,KR,BR,SGR,CGR,SGS,CGS,T1,T2,M)        TKER0010
C   ***    COMPUTES EITHER THE TOTAL KERNELS  (IND=0)  USED IN THE       TKER0012
C          CALCULATION OF A FINITE LENGTH DOUBLET LINE,  OR THE          TKER0014
C          INCREMENTAL OSCILLATORY KERNELS  (IND=1)  USED IN EVALUATING  TKER0016
C          THE INFLUENCE COEFFICIENT MATRIX ELEMENTS                     TKER0018
      REAL   M, KR, KKR, KKI, I00R, I00I, J00R, J00I, I10R, I10I, I20R3,TKER0020
     1       I20I3, I0UR, I0UI, J0UR, J0UI, I1UR, I1UI, I2UR3, I2UI3,    TKER0030
     2       K1, MU1, MU, K2                                             TKER0040
      REAL   K10,K20,K1RT1,K1IT1,K2RT2P,K2IT2P,K10T1,K20T2P              TKER0050
      COMMON  /DLM/  K10,K20,K1RT1,K1IT1,K2RT2P,K2IT2P,K10T1,K20T2P,E2   TKER0060
      COMMON      /KOS/ IND,K01R,K01I,K02R,K02I                          TKER0070
      REAL        K01R,K01I, K02R,K02I                                   TKER0080
   10 FORMAT  (1H0,6E15.6/)                                              TKER0090
      EPS   = 0.00001                                                    TKER0100
      PI = 3.1415926                                                     TKER0110
      PI2= PI*PI                                                         TKER0120
      K10     = 0.0                                                      TKER0130
      K20     = 0.0                                                      TKER0140
      K1RT1   = 0.0                                                      TKER0150
      K1IT1   = 0.0                                                      TKER0160
      K2RT2P  = 0.0                                                      TKER0170
      K2IT2P  = 0.0                                                      TKER0180
      K10T1   = 0.0                                                      TKER0190
      K20T2P  = 0.0                                                      TKER0200
      R1= SQRT (YO*YO+ZO*ZO)                                             TKER0210
      R1S = R1                                                           TKER0220
      IF  (ABS(R1) . GT . EPS)  GO TO  200                               TKER0230
  100 IF  ( XO )                  110,120,120                            TKER0240
  110 KKR=0.                                                             TKER0250
      KKI=0.                                                             TKER0260
      GO TO  995                                                         TKER0270
  120 C1= KR*XO/BR                                                       TKER0280
      T1 = CGR*CGS + SGR*SGS                                             TKER0290
      KKR= 2.*(COS(C1)-1.)*T1                                            TKER0300
      KKI=-2.*SIN(C1)*T1                                                 TKER0310
      K10     = 2.0                                                      TKER0320
      K1RT1   = 2.0*T1*COS(C1)                                           TKER0330
      K1IT1   =-2.0*T1*SIN(C1)                                           TKER0340
      K10T1   = 2.0*T1                                                   TKER0350
      GO TO  995                                                         TKER0360
  200 C1 = CGR                                                           TKER0370
      C2 = SGR                                                           TKER0380
      C3 = CGS                                                           TKER0390
      C4 = SGS                                                           TKER0400
      T2P=(ZO*ZO*C1*C3+YO*YO*C2*C4-ZO*YO*(C2*C3+C1*C4))                  TKER0410
      T2    = (100.*T2P)/(BR*BR)                                         TKER0420
      IF   ( ABS(T2)-EPS )                  210,220,220                  TKER0430
  210 ICHUZ=1                                                            TKER0440
      T1 = CGR*CGS + SGR*SGS                                             TKER0450
      T2 = 0.0                                                           TKER0460
      GO TO 300                                                          TKER0470
  220 T1 = CGR*CGS + SGR*SGS                                             TKER0480
      IF   ( ABS(T1)-EPS )                  230,240,240                  TKER0490
  230 ICHUZ=2                                                            TKER0500
      T1=0.                                                              TKER0510
      GO TO 300                                                          TKER0520
  240 ICHUZ=3                                                            TKER0530
  300 BETA2 = (1.-M*M)                                                   TKER0540
      BIGR = SQRT (XO*XO+BETA2*R1*R1)                                    TKER0550
      K1= KR*R1/BR                                                       TKER0560
      MU1= (M*BIGR-XO)/ (BETA2*R1)                                       TKER0570
      MU=ABS(MU1)                                                        TKER0580
      K2=K1*K1                                                           TKER0590
      IF   ( MU1 )                 310,320,310                           TKER0600
  310 ICHUZ=ICHUZ+3                                                      TKER0610
      GO TO 330                                                          TKER0620
  320 ICHUZ=ICHUZ+6                                                      TKER0630
C     (N*C)**2   FOR   N=1,11   AND C=.372 =                             TKER0640
C                                                                        TKER0650
C       .139384        .553536        1.245456        2.214144           TKER0660
C     3.4596         4.981824       6.780816        8.856576             TKER0670
C    11.209104      13.8384        16.744464                             TKER0680
C                                                                        TKER0690
C     (N*C)   FOR   N=1,12   AND   14,16,18,20,22   =                    TKER0700
C                                                                        TKER0710
C        .744          1.116          1.488           1.86       2.232   TKER0720
C       2.604          2.976          3.348           3.72       4.092   TKER0730
C       4.464          5.208          5.952           6.696      7.44    TKER0740
C       8.184                                                            TKER0750
C                                                                        TKER0760
C     A(N)   FOR N=1,11   =                                              TKER0770
C                                                                        TKER0780
C       .24186198       -2.7918027        24.391079       -111.59196     TKER0790
C   271.43549         -305.75289        -41.13363          545.98537     TKER0800
C  -644.78155          328.72755        -64.279511                       TKER0810
  330 CONTINUE                                                           TKER0820
      FXARG = -0.372*MU                                                  TKER0821
C    **  THE FOLLOWING TEST ON THE SIZE OF THE ARGUMENT TO  EXP  IS      TKER0822
C          NEEDED TO AVOID UNDERFLOW IN  SUBPROGRAM  EXP                 TKER0823
      IF   (FXARG.GE.(-180.0))  GO TO  335                               TKER0824
      E   = 0.0                                                          TKER0825
      GO TO  337                                                         TKER0826
  335 E   = EXP(FXARG)                                                   TKER0827
  337 CONTINUE                                                           TKER0828
                                                                         TKER0830
      C1  = .139384*K2                                                   TKER0840
      C2  = .553536*K2                                                   TKER0850
      C3  = 1.245456*K2                                                  TKER0860
      C4   = 2.214144 *K2                                                TKER0870
      C5   = 3.4596    *K2                                               TKER0880
      C6   = 4.981824 *K2                                                TKER0890
      C7   = 6.780816 *K2                                                TKER0900
      C8   = 8.856576 *K2                                                TKER0910
      C9   = 11.209104*K2                                                TKER0920
      C10  = 13.8384   *K2                                               TKER0930
      C11  = 16.744464*K2                                                TKER0940
                                                                         TKER0950
      R1   = .24186198 / C1                                              TKER0960
      R2   =-2.7918027 / C2                                              TKER0970
      R3   = 24.391079 / C3                                              TKER0980
      R4   =-111.59196 / C4                                              TKER0990
      R5   = 271.43549 / C5                                              TKER1000
      R6   =-305.75289 / C6                                              TKER1010
      R7   =-41.13363  / C7                                              TKER1020
      R8   = 545.98537 / C8                                              TKER1030
      R9   =-644.78155 / C9                                              TKER1040
      R10  = 328.72755 / C10                                             TKER1050
      R11  =-64.279511 / C11                                             TKER1060
```

```
      IF  ( ICHUZ .LT. 4 )                         GO TO 340
      IOOR = .372*(R1 +2.*R2 + 3.*R3 + 4.*R4 + 5.*R5 + 6.*R6 + 7.*R7 +
     1        8.*R8 + 9.*R9 + 10.*R10 + 11.*R11)
      IOOI = -K1*(R1+R2+R3+R4+R5+R6+R7+R8+R9+R10+R11)

  340 GO TO  (420,350,350,390,350,350,380,350,350),ICHUZ

  350 Q1   = R1/ C1
      Q2   = R2/ C2
      Q3   = R3/ C3
      Q4   = R4/ C4
      Q5   = R5/ C5
      Q6   = R6/ C6
      Q7   = R7/ C7
      Q8   = R8/ C8
      Q9   = R9/ C9
      Q10  = R10/C10
      Q11  = R11/C11

      GO TO  (420,410,410,390,360,360,380,360,360),ICHUZ

  360 JOOR =   Q1*(.138384-K2)+Q2*(.553536-K2)+Q3*(1.245456-K2)+Q4*
     1         (2.214144-K2)+Q5*(3.4596-K2)+Q6*(4.981824-K2)+Q7*(6.780816
     2         -K2)+Q8*(8.856576-K2)+Q9*(11.209104-K2)+Q10*(13.8384-K2)+
     3         Q11*(16.744464-K2)
      I2OR3= 2.*K1*IOOI+K2*JOOR

      GO TO  (420,410,410,390,410,390,380,370,370),ICHUZ

  370 JOOI = -K1*(.744*Q1+1.488*Q2+2.232*Q3+2.976*Q4+3.72*Q5+4.464*Q6+
     1              5.208*Q7+5.952*Q8+6.696*Q9+7.44*Q10+8.184*Q11)
      I2OI3= -K1*IOOR+K2*JOOI

      IF  ( ICHUZ .EQ. 8 )                   GO TO 500
  380 I1OI = -K1*IOOR
  390 I1OR = 1.+K1*IOOI

  400 GO TO (420,410,410,420,410,410,500,500,500),ICHUZ

  410 JOUR = E*(Q1*(.138384-K2+.372*MU*C1)+
     1        E*(Q2*(.553536-K2+.744*MU*C2)+
     2        E*(Q3*(1.245456-K2+1.116*MU*C3)+
     3        E*(Q4*(2.214144-K2+1.488*MU*C4)+
     4        E*(Q5*(3.4596-K2+1.86*MU*C5)+
     5        E*(Q6*(4.981824-K2+2.232*MU*C6)+
     6        E*(Q7*(6.780816-K2+2.604*MU*C7)+
     7        E*(Q8*(8.856576-K2+2.976*MU*C8)+
     8        E*(Q9*(11.209104-K2+3.348*MU*C9)+
     9        E*(Q10*(13.8384-K2+3.72*MU*C10)+
     A        E*(Q11*(16.744464-K2+4.092*MU*C11))))))))))))

       JOUI = -K1*(E*(Q1*(.744*MU*C1) + E*(Q2*(1.488*MU*C2) +
     1              E*(Q3*(2.232*MU*C3)+ E*(Q4*(2.976*MU*C4) +
     2              E*(Q5*(3.72*MU*C5) + E*(Q6*(4.464*MU*C6) +
     3              E*(Q7*(5.208*MU*C7) + E*(Q8*(5.952*MU*C8) +
     4              [illegible] + E*(Q10*(7.44*MU*C10) +
     5              [illegible]))))))

  420 IOUR = .372*E*(R1+E*(2.*R2+E*(3.*R3+E*(4.*R4+E*(5.*R5+E*(6.*R6+
     1              E*(7.*R7+E*(8.*R8+E*(9.*R9+E*(10.*R10+E*11.*R11)))))))
     2                 ))))

      IOUI = -K1*(E*(R1+E*(R2+E*(R3+E*(R4+E*(R5+E*(R6+E*(R7+E*(R8+E*(R9
     1              +E*(R10+E*R11))))))))))))
      R1   = R1S
      C6= K1*MU
      C1= SIN(C6)
      C2= COS(C6)
      C3= SQRT(1.+MU*MU)
      C4= MU/C3
      C5= C4/(1.+MU*MU)
      GO TO  (430,440,430,430,440,430,500,500,500),ICHUZ
  430 I1UR = C2*(1.-C4*K1*IOUI)-C1*K1*IOUR
      I1UI =-C2*K1*IOUR-C1*(1.-C4*K1*IOUI)
      GO TO  (500,440,440,460,440,440,500,500,500),ICHUZ
  440 I2UR3 = C2*(2.*(1.-C4)-C5*K1*IOUI+K2*JOUR)+C1*(C6*(1.-C4)-K1*IOUR
     1              +K2*JOUI)
      I2UI3 = C2*(C6*(1.-C4)-K1*IOUR+K2*JOUI)-C1*(2.*(1.-C4)-C5*K1*IOUI
     1              +K2*JOUR)
      GO TO  (500,500,500,460,450,450,500,500,500),ICHUZ
  450 I2UR3 = 2.0 * I2OR3 - I2UR3
      IF  ( ICHUZ-5 )                       500,460,500
  460 CAR = 2.*I1OR-I1UR
      I1UR= CAR
  500 OK1R=0.
      OK1I=0.
      OK2R=0.
      OK2I=0.
      C3=K1*MUI
      C1=COS(C3)
      C2=SIN(C3)
      C3= M*R1/BIGR
      C4=SQRT(1.+MUI*MUI)
      C5=K3*X0/BR
      C6=COS(C5)
      C7=SIN(C5)
      GO TO (530,540,530,530,540,530,510,520,510),ICHUZ
  510 I1UR=I1OR
      I1UI=I1OI
      IF  ( ICHUZ-7 )                       520,530,520
  520 I2UR3= I2OR3
      I2UI3= I2OI3
      IF  ( ICHUZ-8 )                       530,540,530
  530 CK1R = I1UR + C3*C1/C4
      CK1I = I1UI - C3*C2/C4
      K10  = 1.0 + X0/BIGR
      OK1R = CK1R*C6 + CK1I*C7
      OK1I = CK1I*C6 - CK1R*C7
      GO TO  (900,540,540,900,540,540,900,540,540),ICHUZ
  540 C8=  (BETA2*(R1/BIGR)**2 + (2.*MUI*C3)/(C4*C4))*(-C3/C4)
      C9= ( K1*C3)*( C3/C4)
      CK2R = -I2UR3 + C8*C1 - C9*C2
      CK2I = -I2UI3 - C9*C1 - C8*C2
      K20  = -2.0 -X0*(2.0+BETA2*(R1/BIGR)**2)/BIGR
      OK2R = CK2R*C6 + CK2I*C7
      OK2I = CK2I*C6 - CK2R*C7
  900 KKR  = T1*OK1R + T2*OK2R
```

```
      KKI = T1*DK1I + T2*DK2I
      K1RT1  = T1 * DK1R
      K1IT1  = T1 * DK1I
      K2RT2P = T2P* DK2R
      K2IT2P = T2P* DK2I
      K1OT1  = K10* T1
      K2OT2P = K20* T2P
  905 CONTINUE
      KO1R = K1RT1 - K1OT1*FLOAT(IND)
      KO1I = K1IT1
      KO2R = K2RT2P-K2OT2P*FLOAT(IND)
      KO2I = K2IT2P
      RETURN
      END


CTVOR           NORMALWASH AT A POINT (X,Y,Z) - OF A SURFACE DIHEDRAL -
C               DUE TO A TRAPEZOIDAL UNSTEADY VORTEX RING OF UNIT
C               STRENGTH.
C                         SUBROUTINES USED - SNPDF, IDF1, IDF2, FLLD
C
      SUBROUTINE  TVOR  ( SL1, CL1, TL1, SL2, CL2, TL2, SGS, CGS,
     *                    SGR, CGR, XO1, XO2, YO , ZO , E  , BETA,
     *    CHAR,   FMACH,   KR,    BRE,   BIM,
     *    IPRNT )
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C *                                                                  *
C *      SL1, CL1, TL1  SIN(LAMBDA-1), COS(LAMBDA-1), TAN(LAMBDA-1)  *
C *      SL2, CL2, TL2  SIN(LAMBDA-2), .....                         *
C *      SGS, CGS       SIN(GAMMA-S),  .....                         *
C *      SGR, CGR       SIN(GAMMA-R),  ....                          *
C *      X 1            X-XI1                                        *
C *      XO2            X-XI2                                        *
C *      YO             Y - ETA                                      *
C *      ZO             Z - ZETA                                     *
C *      E                                                           *
C *      BETA           SQRT( 1 -FMACH**2 )                          *
C *      CV                                                          *
C *      BR                                                          *
C *      FMACH          MACH NO.                                     *
C *      BRE            REAL PART OF B       (RETURNED)              *
C *      BIM            IMAGINARY PART OF B (RETURNED)               *
C *                                                                  *
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
      REAL     KR, KO1, KO2
C
C               VARIABLES DIMENSIONED (2), FIRST WORD IS THE REAL PART OF
C               THE VALUE AND THE SECOND IS THE IMAGINARY PART
C
      DIMENSION     DKI (2),      DKC (2),       DKO (2),
     *              KO1 (2),      KO2 (2)
      DATA  NPOT /6/
C
C
      DATA  PI43  / 150.79644720 /
C                                                 CALCULATE    3S
C
      L          =      1
      CV         =      XO1 - XO2
      SL      =  SL1
      CL      =  CL1
      TL      =  TL1
      XO      =  XO1
      EE      =  E ** 2
      TE         =   2.0 * E
      ASSIGN  50  TO  ISNP
C                                          CALL  SNPDF
      GO TO  1000
   50 BS         =      DIJ
      SL      =  SL2
      CL      =  CL2
      TL      =  TL2
      XO      =  XO2
      ASSIGN  100  TO  ISNP
C                                          CALL SNPDF
      GO TO     1000
  100 BS         =     3S    -  DIJ
C                                                CALCULATE   DELTA -B
C                   LIMITS FOR SMALL VALUES OF RADII
      EPS         =   0.25 * EE
      IB          =  0
      IC          =   0
      FB          =    1.0
      FC          =    4.0
C          FIRST CALC.
C               DELTA-KD- II, IC, AND ID
C
      FTL1     =    E * TL1
      FTL2     =    E * TL2
      ESGS     =    E * SGS
      ECGS     =    E * CGS
C
      DXO1     =    XO1 + ETL1
      DXO2     =    XO2 + FTL2
      DYO      =    YO  + ECGS
      DZO      =    ZO  + ESGS
C
      ASSIGN    200  TO  IFLLO
C               CALCULATE  R-I  SQUARED AND CALL FLLO IF LARGE ENOUGH
      R2          =     DYO ** 2 +  DZO ** 2
      IF ( R2  .GE.  EPS )   GO TO 2000
      IB          =     1
      FC          =    6.0
      FB          =    0.0
      GO TO 230
  200 DO  220  I = 1, 2
C
  220 DKI(I)    =  KO1(I) / R2  +  KO2(I) / R4
C                              KO1C AND KO2C
  230 DXO1      =     XO1
      DXO2     =   XO2
      DYO      =   YO
      DZO      =   ZO
```

```
      ASSIGN   300  TO  IFLLO                                                TVOR1000
C              CALCULATE  R-C  SQUARED AND CALL FLLO IF LARGE ENOUGH         TVOR1010
      R2        =     DYO ** 2 +  DZO ** 2                                   TVOR1020
      IF ( R2  .GE.  EPS )    GO TO 2000                                     TVOR1030
      IC        =    1                                                       TVOR1040
      FC        =    0.0                                                     TVOR1050
      FB        =    3.0                                                     TVOR1060
      GO TO 330                                                              TVOR1070
  300 DO   320   I  = 1, 2                                                   TVOR1080
C                                                                            TVOR1090
  320 DKC(I) =  KD1(I) / R2   +  KD2(I) / R4                                 TVOR1100
C                           KD10 AND KD20                                    TVOR1110
C              SKIP IF  R-I IS TOO SMALL                                     TVOR1120
  330 IF ( IB .NE. 0 ) GO TO 430                                             TVOR1130
      DX01      =  X01 - ETL1                                                TVOR1140
      DX02      =  X02 - ETL2                                                TVOR1150
      DYO       =  YO  - ECGS                                                TVOR1160
      DZO       =  ZO  - ESGS                                                TVOR1170
C                                                                            TVOR1180
      ASSIGN   400  TO  IFLLO                                                TVOR1190
C              CALCULATE  R-O  SQUARED AND CALL FLLO IF LARGE ENOUGH         TVOR1200
      R2        =     DYO ** 2  +  DZO ** 2                                  TVOR1210
      IF ( R2  .GE.  EPS )    GO TO 2000                                     TVOR1220
      FB        =    0.0                                                     TVOR1230
      FC        =    6.0                                                     TVOR1240
      IB        =    1                                                       TVOR1250
      GO TO 430                                                              TVOR1260
  400 DO   420  I  = 1, 2                                                    TVOR1270
C                                                                            TVOR1280
  420 DKO(I)    =    KD1(I) / R2   +  KD2(I) / R4                            TVOR1290
C                                                                            TVOR1300
  430 COEF      =     1.0 / PI4R                                             TVOR1310
      BRE       =     BS/(TE*CV)- COEF*(FB * (DKI(1)+ DKO(1)) + FC*DKC(1))TVOR1320
      BIM       =               - COEF* ( FB * (DKI(2)+ DKO(2)) + FC*DKC(2)) TVOR1330
C                                                                            TVOR1340
      RETURN                                                                 TVOR1350
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * TVOR1360
C                                                                            TVOR1370
 1000 CALL SNPDF ( SL, CL, TL, SGS, CGS, SGR, CGR, XO, YO, ZO, E,            TVOR1380
     *              GIJ, BETA ,  CV  )                                       TVOR1390
C                             RETURN TO BRANCH                               TVOR1400
      GO TO   ISNP ,  ( 50, 100   )                                          TVOR1410
C                                                                            TVOR1420
C                                                                            TVOR1430
 2000 CALL  FLLO  ( DX01, DX02, DYO, DZO,    SGR, CGR,  SGS,  CGS,          TVOR1440
     *    KR       , CBAR     , FMACH    , F         , L         ,          TVOR1450
     *              KD1(1), KD1(2), KD2(1), KD2(2)  )                        TVOR1460
      IF (  IPRNT  .NE. 0 )  WRITE ( NPOT, 3000)  KD1,  KD2,    R2           TVOR1470
 3000 FORMAT('0KD1=',2E20.8,' KD2=', 2E20.3, ' R2=', E20.3)                  TVOR1480
C                                                                            TVOR1490
      R4        =     R2 * R2                                                TVOR1500
      GO TO IFLLO , ( 200, 300, 400 )                                        TVOR1510
C                                                                            TVOR1520
      END                                                                    TVOR1530
```

```
      SUBROUTINE    WRTFMF  (  IUNT, NBE, FZ, FY, EMZ, EMY )                 WRTF0010
C   ***   WRITES COMPLEX ARRAYS WITH A FORMAT                                WRTF0015
      COMPLEX        FZ   (1), FY   (1), EMZ  (1), EMY  (1)                  WRTF0020
      WRITE ( IUNT, 100 ) (I,  FZ(I), FY(I), EMZ(I), EMY(I), I= 1, NBE) WRTF0030
      RETURN                                                                 WRTF0040
  100 FORMAT  ( I5,  8E12.4  )                                               WRTF0050
      END                                                                    WRTF0060


      SUBROUTINE    WRTFMU(IUNT,MODE,NBE, FZ, FY, EMZ, EMY )                 WRTU0010
C   ***   WRITES COMPLEX ARRAYS WITHOUT A FORMAT                             WRTU0015
      COMPLEX        FZ   (1), FY   (1), EMZ  (1), EMY  (1)                  WRTU0020
      WRITE ( IUNT ) MODE, NBE, ( FZ(I), FY(I), EMZ(I), EMY(I),I=1,NBE) WRTU0030
      RETURN                                                                 WRTU0040
      END                                                                    WRTU0050
```

```
      SUBROUTINE DATA                                                   DATA0010
C *** CALCULATES ARRAYS OF GEOMETRY FOR ALL LIFTING SURFACES            DATA0020
C     AND ALL INTERFERENCE- AND SLENDER BODIES                          DATA0030
      COMMON   NP,NTP,NB,NBZ,NBY,NTZ,NTY,FMACH,REFA,REFS,REFC,ND,NE,    DATA0040
     1         NBEA(10,2),NSBEA(10),NCARAY(100),NSARAY(100),NBARAY(100)DATA0050
     2         ,ZB(10),YB(10),XLE(10),XTE(10),ARB(10),AVR(10),AC(200),  DATA0060
     3         AOP(200),XIS1(200),XIS2(200),XIJ(100),CS(210),YS(210),   DATA0070
     4         ZS(210),FE(200),SG(200),CG(200),X(500),XIC(500),KR,XM,   DATA0080
     5         DELX(500),XLAM(500),RIA(100),THIA(100),TH2A(100),NFL(10)DATA0090
     6         ,IFLA(30,2),NTI2(10,2),NAS(100),NAS3(200),YIN(100),      DATA0100
     7         ZIN(100),DETA(50),DZET(50),NOBODY(50),NPANEL(50),NTO     DATA0110
     8         ,NTZS,NTYS                                               DATA0120
      DIMENSION  NAB(10),KTF(100),NTH(100),THE(50),TAU(50),GMA(50),     DATA0130
     1         XII(100),XIS(100),RS(100),XS(100),ETA(100),ZET(100),     DATA0140
     2         TH1(24), TH2(24), RI(100),YS1(200),ZS1(200),YS2(200),    DATA0150
     3         ZS2(200),DYS(200),DZS(200),GAR(210),XI1(500),XI2(500)   DATA0160
      REAL      KR                                                      DATA0170
   10 FORMAT  (6F10.0)                                                  DATA0180
   20 FORMAT  (1H0,3X,101H**** NUMBER OF ASSOCIATED BODY IS GREATER THANDATA0190
     1 NUMBER OF BODIES  --  INPUT ERROR   *   IGNORE BODY  //)         DATA0200
   30 FORMAT  (1H0,30X,I4,31H  CHORDWISE DIVISIONS FOR PANEL, I3//)     DATA0210
   40 FORMAT  (2F10.0,2I10,10I2)                                        DATA0220
   50 FORMAT  (1H0,30X,I4,31H  SPANWISE  DIVISIONS FOR PANEL, I3//)     DATA0230
   60 FORMAT  (4F10.0,2I3,7I2)                                          DATA0240
   70 FORMAT  (1H0,10X,6E19.8)                                          DATA0250
   80 FORMAT  (6I10)                                                    DATA0260
   90 FORMAT  (1H0,30X,13H**  PANEL NO.,I4,17H INPUT VALUES  **//)      DATA0270
  100 FORMAT  (4F10.0,2I3,7I2)                                          DATA0280
  110 FORMAT (1H0,5X,4HX1 =,F12.6,6H  X2 =,F12.6, 6H  Y1 =,F12.6, 6H  Z1DATA0290
     1 =,F12.6//5X,4HX3 =,F12.6,6H  X4 =,F12.6,6H  Y2 =,F12.6,5H  Z2 =, DATA0300
     2        F12.6//6X,4HNC =,I3,6H  NS =,I3//6X,19HASSOCIATED BODIES =DATA0310
     3        10I5//)                                                   DATA0320
  120 FORMAT (1H0,25X,12H** BODY NO.,I4,19H  INPUT VALUES **// 15X,     DATA0330
     1 32HCENTER OF BODY COORDINATES   Y =, F16.6,5X, 3HZ =, F16.6/ 15X,DATA0340
     2 29HAVERAGE HALF-WIDTH OF BODY  =,  F16.6/ 15X, 30HCROSS-SECTIONALDATA0350
     3 ASPECT RATIO =,F15.6/ 15X, 41HNUMBER OF INTERFERENCE ELEMENTS ON DATA0360
     4BODY =,I4/ 15X,33HNUMBER OF SLENDER BODY ELEMENTS =,I4/ 15X,22HZ-YDATA0370
     5 ORIENTATION FLAG =, I4/ 15X, 9HWI-FLAG =, I3,  12H  R-S FLAG =,I3DATA0380
     6/ 15X,38HNUMBER OF DELTA-ETA DELTA-ZETA PAIRS =,I4//)             DATA0390
  130 FORMAT  (1H0//17X,4HBODY,5X, 5HPANEL,6X, 9HDELTA-ETA,6X,10HDELTA-ZDATA0400
     1ETA/)                                                             DATA0410
  140 FORMAT  (6I10)                                                    DATA0420
  150 FORMAT  (2(I10,2F10.0))                                           DATA0430
  160 FORMAT  (1H ,10X,2I10,2F16.6)                                     DATA0440
  17  FORMAT  (1H1,25X,29H** SUMMARY OF PANEL DATA  **// 7X,50HPANEL    DATA0450
     1NC   NS   NO-ARRAY   DIHEDRAL  NO. OF ASSOC.,3X,2*HLIST OF ASSOCIATEDATA0460
     2D BODIES/35X, 1H0ANGLE          BODIES/)                           DATA0470
  180 FORMAT  (1H1)                                                     DATA0480
  190 FORMAT  (1H0,30X,I4,24H  XI-I ELEMENTS FOR BODY, I3//)            DATA0490
  200 FORMAT  (1H0,30X,I4,24H  S-I  ELEMENTS FOR BODY, I3//)            DATA0500
  210 FORMAT  (1H ,6X,3I5,I4,F12.5,7X,I3,7X,10I4)                       DATA0510
  220 FORMAT  (1H1,33X,39H** GEOMETRY ARRAYS FOR ALL PANELS  **//5X,57HDATA0520
     1PANEL  STRIP  BOX   3/4 CHORD       1/4 CHORD X-COORDINATES, 8X,  DATA0530
     2LIH  BOX CHORD,4X, 9H1/4 CHORD/ 7X,15HNO.   NO.   NO.,7X,1HX,7X,  DATA0540
     3INBOARD       CENTER       OUTBOARD       DELTA-X      SWEEP ANGLE/)DATA0550
  230 FORMAT  (1H0,30X,I4,24H  XI-S ELEMENTS FOR BODY, I3//)            DATA0560
  240 FORMAT  (1H0,30X,I4,24H  R-S  ELEMENTS FOR BODY, I3//)            DATA0570
  250 FORMAT  (1H0,30X,I4,30H  DELTA-ETA  ELEMENTS FOR BODY, I3//)      DATA0580
  260 FORMAT  (1H0,4X,3I5,4F12.5)                                       DATA0590
  270 FORMAT  (1H0,30X,I4,27H  THETA-1 ELEMENTS FOR BODY, I3//)         DATA0600
  280 FORMAT  (1H0,30X,I4,27H  THETA-2 ELEMENTS FOR BODY, I3//)         DATA0610
  290 FORMAT  (1H0,30X,I4,30H  DELTA-ZETA ELEMENTS FOR BODY, I3//)      DATA0620
  300 FORMAT  (1H1,5X, 9HSTRIP NO.,7X, 1HY,11X,1HZ,7X,19HDELTA-Y     DELDATA0630
     1TA-Z,7X, 1HF, 1LX,19HCHORD            X-L.E.//)                   DATA0640
  310 FORMAT  (1H0,30X,46H** RECEIVING POINT ARRAYS  X, Y, Z, GAMMA **//)DATA0650
  320 FORMAT  (1H0,30X,49H** SENDING POINT ARRAYS XI1, XI2, AO, AO-PRIMEDATA0660
     1 **/)                                                             DATA0670
  330 FORMAT  (1H0,10X,I3,7F12.5)                                       DATA0680
  340 FORMAT  (1H0,10X,53HTHE FIRST AND LAST BODY ELEMENTS FOR THETA-I  DATA0690
     1ON BODY, I3//)                                                    DATA0700
  350 FORMAT  (1H0,3 X,10I5/)                                           DATA0710
  360 CONTINUE                                                          DATA0720
      PI  = 3.1415926                                                   DATA0730
      NATO  = 0                                                         DATA0740
      NSTO  = 0                                                         DATA0750
      N     = 0                                                         DATA0760
      JS    = 0                                                         DATA0770
      LAS   = 0                                                         DATA0780
      DO 370  I=1,24                                                    DATA0790
      TH1(I)     = 0.0                                                  DATA0800
      TH2(I)     = 0.0                                                  DATA0810
  370 CONTINUE                                                          DATA0820
      DO  380  I=1,50                                                   DATA0830
      NOBODY(I) = 0                                                     DATA0840
      NPANEL(I) = 0                                                     DATA0850
      DETA(I)    = 0.0                                                  DATA0860
      DZET(I)    = 0.0                                                  DATA0870
  380 CONTINUE                                                          DATA0880
      DO  390  I=1,100                                                  DATA0890
      NCARAY(I)  = 0                                                    DATA0900
      NSARAY(I)  = 0                                                    DATA0910
      NBARAY(I)  = 0                                                    DATA0920
      NBEA(I,1)  = 0                                                    DATA0930
      NBEA(I,2)  = 0                                                    DATA0940
      NSBEA(I)   = 0                                                    DATA0950
      XIJ(I)     = 0.0                                                  DATA0960
      RIA(I)     = 0.0                                                  DATA0970
      THIA(I)    = 0.0                                                  DATA0980
      TH2A(I)    = 0.0                                                  DATA0990
      NAS(I)     = 0                                                    DATA1000
      KTF(I)     = 0                                                    DATA1010
      XII(I)     = 0.0                                                  DATA1020
      XIS(I)     = 0.0                                                  DATA1030
      RS(I)      = 0.0                                                  DATA1040
      XS(I)      = 0.0                                                  DATA1050
      NTH(I)     = 0                                                    DATA1060
      ETA(I)     = 0.0                                                  DATA1070
      ZET(I)     = 0.0                                                  DATA1080
      RI(I)      = 0.0                                                  DATA1090
  390 CONTINUE                                                          DATA1100
      DO 400  I=1,200                                                   DATA1110
      AO(I)      = 0.0                                                  DATA1120
      AOP(I)     = 0.0                                                  DATA1130
      XIS1(I)    = 0.0                                                  DATA1140
      XIS2(I)    = 0.0                                                  DATA1150
  400 NASB(I) = 0                                                       DATA1160
      DO 460  LP=1,NP                                                   DATA1170
      WRITE (6,90)  LP                                                  DATA1180
      J      = 0                                                        DATA1190
C                                                                       DATA1200
C **  READ PANEL COORDINATES, NUMBER OF CHORDWISE BOXES AND (SPANWISE)DATA1210
C     STRIPS,  AND THE ASSOCIATED BODIES                                DATA1220
C                                                                       DATA1230
      READ  (5,10)   X1,X2,X3,X4,Y1,Y2                                  DATA1240
      READ  (5,40)   Z1,Z2,NC,NS,(NAB(I), I=1, 6)                       DATA1250
      WRITE (6,110)  X1,X2,Y1,Z1, X3,X4,Y2,Z2, NC,NS,(NAB(I), I=1, 6)   DATA1260
C                                                                       DATA1270
C **  STORE  Y- AND Z-COORDINATE OF INBOARD EDGE OF PANEL IN THE        DATA1280
C     ARRAYS  YIN(100),   AND  ZIN(100)  RESPECTIVELY                   DATA1290
C                                                                       DATA1300
      YIN(LP) = Y1                                                      DATA1310
      ZIN(LP) = Z1                                                      DATA1320
      NAT   = 0                                                         DATA1330
      IF  (NB.EQ.0)  GO TO  430                                         DATA1340
      DO 420  NA=1, 6                                                   DATA1350
      IF  (NAB(NA).EQ.0)   GO TO 420                                    DATA1360
      IF  (NAB(NA).GT.NB)  GO TO 410                                    DATA1370
      NAT   = NAT+1                                                     DATA1380
      LAS   = LAS+1                                                     DATA1390
      NASB(LAS) = NAB(NA)                                               DATA1400
C **  NASB(200)   IS THE ARRAY OF ASSOCIATED BODIES FOR ALL PANELS      DATA1410
      GO TO  420                                                        DATA1420
  410 CONTINUE                                                          DATA1430
      WRITE  (6,20)                                                     DATA1440
  420 CONTINUE                                                          DATA1450
      NAS(LP) = NAT                                                     DATA1460
C  ** NAS(100)    IS THE ARRAY OF THE NUMBER OF ASSOCIATED BODIES PER   DATA1470
C  PANEL  --  THE MAXIMUM NUMBER OF ASSOCIATED BODIES PER PANEL IS  10  DATA1480
  430 CONTINUE                                                          DATA1490
      NCNS = NC*NS                                                      DATA1500
      NCARAY(LP) = NC                                                   DATA1510
      NSARAY(LP) = NS                                                   DATA1520
      NBARAY(LP) = NATO+NCNS                                            DATA1530
      NSTO = NSTO+NS                                                    DATA1540
      NCP1 = NC+1                                                       DATA1550
      NSP1 = NS+1                                                       DATA1560
C                                                                       DATA1570
C **  READ THE FRACTIONAL CHORDWISE-  AND SPANWISE DIVISIONS FOR        DATA1580
C     PANEL  --  ARRAYS  THE(50),  AND TAU(50)  RESPECTIVELY            DATA1590
```

```
C
      READ  (5,10)  (TH(I),  I=1,NCP1)
      READ  (5,10)  (TAU(I), I=1,NSP1)
      WRITE (6,30)  NCP1,LP
      WRITE (6,70)  (TH(I),  I=1,NCP1)
      WRITE (6,50)  NSP1,LP
      WRITE (6,70)  (TAU(I), I=1,NSP1)
C
C  **   COMPUTE DIHEDRAL ANGLE FOR PANEL  --  ARRAY NAME IS  GMA(10)
C
      YDIF = Y2-Y1
      ZDIF = Z2-Z1
      CALL         ATAN3(ZDIF,YDIF,ATANA)
      GMA(LP) = ATANA*180./PI
      FOUR = X4-X3-X2+X1
      DIF31= X3-X1
      DIF21= X2-X1
      DIF43= X4-X3
      DO 450  J=1,NS
      JS    = JS+1
      JP1   = J +1
      FACT = 0.5*(TAU(J)+TAU(JP1))
      TDIF = TAU(JP1) - TAU(J)
      DO 440  I=1,NC
      IP1   = I+1
      K     = K+1
      TH13 = 0.25*TH(I) + 0.75*TH(IP1)
      TH31 = 0.75*TH(I) + 0.25*TH(IP1)
C
C  **   COMPUTE THE  3/4-  AND  1/4-CHORD X-COORDINATES OF ALL PANEL
C       BOXES  *  ALSO, THE AVERAGE BOX-CHORD  --  STORE THEM IN THE
C       ARRAYS  X(500), XI1(500), XI2(500) AND DELX(500) RESPECTIVELY
C
      X(K)   = FACT    * (TH13*FOUR+DIF31) + TH13*DIF21 + X1
      XI1(K) = TAU(J)  * (TH31*FOUR+DIF31) + TH31*DIF21 + X1
      XI2(K) = TAU(JP1)* (TH31*FOUR+DIF31) + TH31*DIF21 + X1
      DELX(K)= FACT * (TH(IP1)-TH(I))*FOUR + (TH(IP1)-TH(I))*DIF21
  440 CONTINUE
C
C  **   COMPUTE DATA ARRAYS FOR ALL STRIPS OF PANEL  --
C       YS(200)     =  Y-COORDINATE OF CENTERLINE OF STRIP
C       ZS(200)     =  Z-COORDINATE OF CENTERLINE OF STRIP
C       DYS(200)    =  DELTA-Y  OF STRIP
C       DZS(200)    =  DELTA-Z  OF STRIP
C       EE(200)     =  SEMI-WIDTH OF STRIP
C       CS(200)     =  CHORDLENGTH OF STRIP
C       SG(200)     =  SINE   OF THE DIHEDRAL ANGLE OF STRIP
C       CG(200)     =  COSINE OF THE DIHEDRAL ANGLE OF STRIP
C       XIJ(200)    =  X-COORDINATE OF LEADING EDGE OF STRIP
C       GMAR(210)   =  DIHEDRAL ANGLE OF STRIP IN RADIANS
C
      YS(JS) = FACT * YDIF + Y1
      ZS(JS) = FACT * ZDIF + Z1
      DYS(JS)= TDIF * YDIF
      DZS(JS)= TDIF * ZDIF
      EE(JS) = 0.5*SQRT( DYS(JS)**2 + DZS(JS)**2)
      CS(JS) = FACT * FOUR + DIF21
      SG(JS) = DZS(JS)/(2.0*EE(JS))
      CG(JS) = DYS(JS)/(2.0*EE(JS))
      XIJ(JS)= FACT * DIF31 + X1
      GMAR(JS) = GMA(LP)*PI/180.0
  450 CONTINUE
      NSTO = NSARAY(LP)
  460 CONTINUE
      NBOX = NBARAY(NP)
      NSTRIP = NSTO
      WRITE (6,170)
      I1    = 0
      I2    = 0
      DO 470  L = 1,NP
      I1    = I2+1
      I2    = I2+NAS(L)
C  DIHEDRAL ANGLE PER PANEL IS GIVEN IN DEGREES
      WRITE (6,210)  L,NCARAY(L),NSARAY(L),NBARAY(L),GMA(L),NAS(L),
     1                   (NASR(I),  I=I1,I2)
  470 CONTINUE
      WRITE (6,220)
      LP    = 1
      J     = 1
      NBXP = NBARAY(1)
      NBXS = NCARAY(1)
      SIGN = 1.0
      DO 500  I=1,NBOX
      IF (I.LE.NBXP)  GO TO 480
      LP    = LP+1
      NBXP = NBARAY(LP)
  480 CONTINUE
      IF (I.LE.NBXS)  GO TO 490
      J     = J+1
      NBXS = NBXS+NCARAY(LP)
  490 CONTINUE
      XIC(I)  =  0.5*(XI1(I)+XI2(I))
      UP    = XI2(I)-XI1(I)
      DOWN = 2.0*EE(J)*SIGN
      CALL         ATAN3(UP,DOWN,ATANX)
      XLAM(I) = ATANX
      WRITE (6,260)  LP,J,I,X(I),XI1(I),XIC(I), XI2(I), DELX(I),XLAM(I)
C  SWEEP ANGLE IS WRITTEN IN RADIANS,  BUT IT IS STORED IN  XLAM(500) AS
C  TANGENT LAMBDA
      XLAM(I) = UP/DOWN
  500 CONTINUE
      WRITE (6,300)
      DO 510  J=1,NSTRIP
      WRITE (6,330)  J,YS(J),ZS(J),DYS(J),DZS(J),EE(J),CS(J),XIJ(J)
  510 CONTINUE
      WRITE (6,180)
      I2    = NBOX
      NTP   = NBOX
      J     = NSTRIP
      NT1   = NBOX
      KT    = 0
      KT1   = 0
      KT    = 0
      LCUM = 0
      IS2   = 0
      LBN   = 0
      NBZ   = 0
      NBY   = 0
      NTZ   = 0
      NTY   = 0
      NTZS = 0
      NTYS = 0
      IT12 = 0
      IT22 = 0
      IXX   = 0
C  **   PANEL GEOMETRY COMPUTATIONS COMPLETE --  BYPASS BODY  DATA-READ
C       AND ARRAY FORMATIONS  IF  NB=0
C
      IF (NB.EQ.0)  GO TO 750
      DO 730  LB=1,NB
C
C  **   READ BODY DATA
C
      READ (5,100)  ZC,YC,RAD,AR,NBE,NSBE,NZY,NRI,NRS,NSH,NT1,NT2
      WRITE (6,120)  LB,YC,ZC,RAD,AR,NBE,NSBE,NZY,NRI,NRS,NSH
C  **   NZY=1  MEANS  BODY IS Z-ORIENTED
C  **   NZY=2  MEANS  BODY IS BOTH Z-  AND Y-ORIENTED
C  **   NZY=3  MEANS  BODY IS Y-ORIENTED
C  **   NOTE THAT BODY DATA HAS TO BE INPUT IN THIS ORDER,  I.E.
C  Z-ORIENTED BODIES FIRST,  Z&Y ORIENTED BODIES NEXT,  AND  Y-ORIENTED
C  BODIES LAST
      NTO   = NTO + NBE
      GO TO (520,520,530), NZY
  520 CONTINUE
      NBZ   =NBZ+1
      NTZ   = NTZ +NBE
      NTZS = NTZS+NSBE
      IF (NZY.EQ.1)  GO TO 540
  530 CONTINUE
      NBY   = NBY+1
      NTY   = NTY +NBE
      NTYS = NTYS+NSBE
  540 CONTINUE
      LBN   = LBN+1
C
C  **   SAVE BODY ARRAYS  --
C       ZB(10)       =  Z-COORDINATE OF BODY CENTER
C       YB(10)       =  Y-COORDINATE OF BODY CENTER
```

```
C        AVR(10)       =  AVERAGE SEMI-WIDTH OF BODY                        DATA3180
C        ARH(10)       =  CROSS SECTIONAL ASPECT RATIO OF BODY              DATA3190
C        NBEA(10,1)    =  NUMBER OF INTERFERENCE BODY ELEMENTS              DATA3200
C        NBEA(10,2)    =  Z-Y (ORIENTATION) FLAG FOR BODY                   DATA3210
C        NSBEA(10)     =  NUMBER OF SLENDER BODY ELEMENTS                   DATA3220
C                                                                           DATA3230
      ZB( LBN) = ZC                                                         DATA3240
      YB( LBN) = YC                                                         DATA3250
      AVR(LBN) = RAD                                                        DATA3260
      ARB(LBN) = AR                                                         DATA3270
      NBEA(LBN,1) = NBE                                                     DATA3280
      NBEA(LBN,2) = NZY                                                     DATA3290
      NSBEA(LBN) = NSBE                                                     DATA3300
      NBEP1= NBE+1                                                          DATA3310
C                                                                           DATA3320
C  **   IF NBE = 0  (NO INTERFERENCE BODY ELEMENTS),  BYPASS READING OF     DATA3330
C       ALL INTERFERENCE BODY ARRAYS                                        DATA3340
C                                                                           DATA3350
      IF (NBE.EQ.0)  GO TO  580                                             DATA3360
C                                                                           DATA3370
C  **   READ  X-COORDINATES OF INTERFERENCE BODY ELEMENT ENDPOINTS          DATA3380
C                                                                           DATA3390
      READ  (5,10)  (XI(I), I=1,NBEP1)                                      DATA3400
      WRITE (6,190)  NBEP1, LB                                              DATA3410
      WRITE (6,70)  (XI(I),   I=1,NBEP1)                                    DATA3420
      IF (NRI.NE.0)  GO TO 550                                              DATA3430
      DO 550  I=1,NBEP1                                                     DATA3440
      RI(I)= RAD                                                            DATA3450
  550 CONTINUE                                                              DATA3460
      GO TO  570                                                            DATA3470
  560 CONTINUE                                                              DATA3480
C                                                                           DATA3490
C  **   READ  RADII  OF INTERFERENCE BODY ELEMENT ENDPOINTS  --  ONLY       DATA3500
C       IF  NRI = 1                                                         DATA3510
C                                                                           DATA3520
      READ  (5,10)  (RI(I),  I=1,NBEP1)                                     DATA3530
  570 CONTINUE                                                              DATA3540
      WRITE (6,230)  NBEP1, LB                                              DATA3550
      WRITE (6,70)  (RI(I),    I=1,NBEP1)                                   DATA3560
C                                                                           
  580 CONTINUE                                                              DATA3570
      NSBEP1 = NSBE+1                                                       DATA3580
C                                                                           DATA3590
C  **   IF NSBE = 0  (NO SLENDER BODY ELEMENTS ),  BYPASS READING OF        DATA3600
C       ALL SLENDER BODY ARRAYS                                             DATA3610
C                                                                           DATA3620
      IF (NSBE.EQ.0) GO TO  620                                             DATA3630
C                                                                           DATA3640
C  **   READ  X-COORDINATES OF SLENDER BODY ELEMENT ENDPOINTS               DATA3650
C                                                                           DATA3660
      READ  (5,10)  (XS(I),   I=1,NSBEP1)                                   DATA3670
      WRITE (6,210)  NSBEP1,LB                                              DATA3680
      WRITE (6,70)  (XS(I),   I=1,NSBEP1)                                   DATA3690
      IF (NRS.NE.0)  GO TO 600                                              DATA3700
      DO 590  I=1,NSBEP1                                                    DATA3710
      RS(I)= RAD                                                            DATA3720
  590 CONTINUE                                                              DATA3730
      GO TO  610                                                            DATA3740
  600 CONTINUE                                                              DATA3750
C                                                                           DATA3760
C  **   READ  RADII  OF SLENDER BODY ELEMENT ENDPOINTS  --  ONLY            DATA3770
C       IF  NRS = 1                                                         DATA3780
C                                                                           DATA3790
      READ  (5,10)  (RS(I), I=1,NSBEP1)                                     DATA3800
  610 CONTINUE                                                              DATA3810
      WRITE (6,240)  NSBEP1,LB                                              DATA3820
      WRITE (6,70)  (RS(I),  I=1,NSBEP1)                                    DATA3830
  620 CONTINUE                                                              DATA3840
      IF (NBE.EQ.0)  GO TO  560                                             DATA3850
C                                                                           DATA3860
C  **   READ  THETA-1-MU  =  ANGULAR ORIENTATION OF POINT  MU  ON BODY      DATA3870
C       --  INPUT  THETA-1-MU  IN DEGREES --                                DATA3880
C       --  SAVE CONTINUOUS ARRAY FOR ALL BODIES IN THIA(100) IN RADIANSDATA3890
C                                                                           DATA3900
      READ  (5,10)  (TH1(I),  I=1,NT1)                                      DATA3910
      WRITE (6,270)  NT1,   LB                                              DATA3920
      WRITE (6,70)  (TH1(I),      I=1,NT1)                                  DATA3930
C                                                                           DATA3940
C  **   BYPASS READING  THETA-2-MU  IF  NT2 = 0                             DATA3950
C                                                                           DATA3960
      IF (NT2.EQ.0)  GO TO  630                                             DATA3970
C                                                                           DATA3980
C  **   READ  THETA-2-MU  =  ANGULAR ORIENTATION OF POINT  MU  ON BODY      DATA3990
C       *  SECOND SET OF VALUES  *  THE SPARSE ONE  *  INPUT IN DEGREES     DATA4000
C       --  SAVE CONTINUOUS ARRAY FOR ALL BODIES IN TH2A(100) IN RADIANSDATA4010
C                                                                           DATA4020
      READ  (5,10)  (TH2(I),  I=1,NT2)                                      DATA4030
      WRITE (6,280)  NT2,   LB                                              DATA4040
      WRITE (6,70)  (TH2(I),     I=1,NT2)                                   DATA4050
  630 CONTINUE                                                              DATA4060
      IA    = 0                                                             DATA4070
      IT11 = IT12+1                                                         DATA4080
      IT12 = IT12+NT1                                                       DATA4090
      DO 640  IT=IT11,IT12                                                  DATA4100
      IA    = IA+1                                                          DATA4110
      THIA(IT) = TH1(IA)*PI/180.0                                           DATA4120
  640 CONTINUE                                                              DATA4130
      IA    = 0                                                             DATA4140
      IT21 =IT22+1                                                          DATA4150
      IT22 = IT22+NT2                                                       DATA4160
      DO 650  IT=IT21,IT22                                                  DATA4170
      IA    = IA+1                                                          DATA4180
      TH2A(IT) = TH2(IA)*PI/180.0                                           DATA4190
  650 CONTINUE                                                              DATA4200
      KT1  = 0                                                              DATA4210
C                                                                           DATA4220
C  **   READ SETS OF  FIRST- AND LAST  INTERFERENCE BODY ELEMENTS           DATA4230
C       ASSSOCIATED WITH  THETA-1-MU  --  SAVE THE NUMBER OF PAIRS FOR      DATA4240
C       BODY IN ARRAY  NFL(10),   AND SAVE CONTINUOUS ARRAYS OF FIRST-      DATA4250
C       AND  LAST  BODY ELEMENTS IN  IFLA(30,1)  AND  IFLA(30,2)            DATA4260
C       RESPECTIVELY                                                        DATA4270
C                                                                           DATA4280
      READ  (5,80)  L1,L2,L3,L4,L5,L6                                       DATA4290
      KT    = KT+1                                                          DATA4300
      KT1  = KT1+1                                                          DATA4310
      IFLA(KT,1) = L1                                                       DATA4320
      IFLA(KT,2) = L2                                                       DATA4330
      IF (L3.EQ.0)  GO TO 660                                               DATA4340
      KT    = KT+1                                                          DATA4350
      KT1  = KT1+1                                                          DATA4360
      IFLA(KT,1) = L3                                                       DATA4370
      IFLA(KT,2) = L4                                                       DATA4380
      IF (L5.EQ.0)  GO TO 660                                               DATA4390
      KT    = KT+1                                                          DATA4400
      KT1  = KT1+1                                                          DATA4410
      IFLA(KT,1) = L5                                                       DATA4420
      IFLA(KT,2) = L6                                                       DATA4430
  660 CONTINUE                                                              DATA4440
C                                                                           DATA4450
C  **   BYPASS  READING  Y-  AND  Z-SHIFT  ELEMENTS  IF  NSH = 0            DATA4460
C                                                                           DATA4470
      IF (NSH.EQ.0)  GO TO 680                                              DATA4480
C                                                                           DATA4490
C  **   READ  NSH  SETS OF Y-  AND  Z-SHIFT ELEMENETS  FOR BODY  --         DATA4500
C       SAVE THE FOLLOWING RELATED INFORMATION  --  NOBODY(50) = BODY       DATA4510
C       NUMBER,  NPANEL(50) = PANEL NUMBER,  DETA(50) = DELTA-ETA =         DATA4520
C       Y-SHIFT OF PANEL FOR BODY,  DZET(50) = DELTA-ZETA = Z-SHIFT OF      DATA4530
C       PANEL FOR BODY                                                      DATA4540
C                                                                           DATA4550
      DO  670  ISH=1,NSH,2                                                  DATA4560
      READ  (5,150)  NC01,C02,C03, NC04,C05,C06                             DATA4570
      IXX   = IXX+1                                                         DATA4580
      NOBODY(IXX) = LB                                                      DATA4590
      NPANEL(IXX) = NC01                                                    DATA4600
      DETA(IXX)    = C02                                                    DATA4610
      DZET(IXX)    = C03                                                    DATA4620
      IF (NC04.EQ.0.OR.ISH.EQ.NSH)  GO TO  670                              DATA4630
      IXX   = IXX+1                                                         DATA4640
      NOBODY(IXX) = LB                                                      DATA4650
      NPANEL(IXX) = NC04                                                    DATA4660
      DETA(IXX)    = C05                                                    DATA4670
      DZET(IXX)    = C06                                                    DATA4680
  670 CONTINUE                                                              DATA4690
  680 CONTINUE                                                              DATA4700
      IF (NBE.EQ.0)  GO TO  700                                             DATA4710
      NFL(LBN)= KT1                                                         DATA4720
      LCUM = LCUM+NBE                                                       DATA4730
      NT12(LBN,1) = NT1                                                     DATA4740
      NT12(LBN,2) = NT2                                                     DATA4750
```

```
      WRITE (6,340) L8                                                  DATA4760
      WRITE (6,350) L1,L2,L3,L4,L5,L6                                   DATA4770
      I1    = I2+1                                                      DATA4780
      I2    = I2+NBE                                                    DATA4790
      K     = 0                                                         DATA4800
C                                                                       DATA4810
C  **   APPEND THE X-, XI1-, XI2-, DELX-, YS-, ZS- AND GMAR-ARRAYS  TO DATA4820
C        END UP WITH CONTINUOUS ARRAYS FOR ALL RECEIVING POINTS  --  TO DATA4830
C        BE WRITTEN ON TAPE  NTPR                                       DATA4840
C                                                                       DATA4850
      DO 690  I=I1,I2                                                   DATA4860
      K     = K+1                                                       DATA4870
      KP1   = K+1                                                       DATA4880
      KI    = KI+1                                                      DATA4890
      IP1   = I+1                                                       DATA4900
      X(I)       = 0.5*(XII(KP1)+XII(K))                                DATA4910
      XI1(I)     = XII(K)                                               DATA4920
      XI2(I)     = XII(KP1)                                             DATA4930
      DELX(I)    = XI2(I)-XI1(I)                                        DATA4940
C                                                                       DATA4950
C  **   COMPUTE RADII OF ALL INTERFERENCE BODY ELEMENT MIDPOINTS  --    DATA4960
C        SAVE CONTINUOUS ARRAY FOR ALL BODIES IN  RIA(100)              DATA4970
C                                                                       DATA4980
      RIA(KI)    = 0.5*(RI(K) + RI(KP1))                                DATA4990
  690 CONTINUE                                                          DATA5000
      J       = J+1                                                     DATA5010
      YS(J)      = YC                                                   DATA5020
      ZS(J)      = ZC                                                   DATA5030
C                                                                       DATA5040
C  **   SAVE LEADING EDGE-  AND TRAILING EDGE X-COORDINATES OF ALL      DATA5050
C        INTERFERENECE BODIES IN THE ARRAYS  XLE(10)  AND  XTE(10)      DATA5060
C        RESPECTIVELY                                                   DATA5070
      XLE(LBN)   = XII(1)                                               DATA5080
      XTE(LBN)   = XII(NBEP1)                                           DATA5090
      CS(J)      = XTE(LBN)-XLE(LBN)                                    DATA5100
      GMAR(J) = 0.0                                                     DATA5110
      IF (NZY.EQ.3)  GMAR(J) =-PI/2.0                                   DATA5120
  700 CONTINUE                                                          DATA5130
      IF (NSBE.EQ.0) GO TO  730                                         DATA5140
      IF  ( NBE.NE.0)  GO TO  710                                       DATA5150
C  **   SAVE LEADING EDGE-  AND TRAILING EDGE X-COORDINATES OF ALL      DATA5160
C        SLENDER        BODIES IN THE ARRAYS  XLE(10)  AND  XTE(10)     DATA5170
C        RESPECTIVELY  --  ONLY IF   NBE = 0  (SEE INPUT)               DATA5180
      XLE(LBN)   = XIS(1)                                               DATA5190
      XTE(LBN)   = XIS(NSBEP1)                                          DATA5200
  710 CONTINUE                                                          DATA5210
      IS1   = IS2+1                                                     DATA5220
      IS2   = IS2+NSBE                                                  DATA5230
      KS    = 0                                                         DATA5240
C                                                                       DATA5250
C  **   SAVE THE FOLLOWING CONTINUOUS ARRAYS FOR ALL  SLENDER BODIES  --DATA5260
C        XIS1(100) = X-COORDINATE OF LEADING EDGE OF SLENDER BODY ELEM.  DATA5270
C        XIS2(100) = X-COORDINATE OF TRAILING EDGE OF SLENDER BODY ELEM. DATA5280
C        AO(100)    = AVERAGE RADIUS OF SLENDER BODY ELEMENT (Y-DIRECTION)DATA5290
C        AOP(100)   = DERIVATIVE OF THE LOCAL AVERAGE RADIUS  AO         DATA5300
C                                                                       DATA5310
      DO 720  IS=IS1,IS2                                                DATA5320
      KS    = KS+1                                                      DATA5330
      KSP1 = KS+1                                                       DATA5340
      XIS1(IS) = XIS(KS)                                                DATA5350
      XIS2(IS) = XIS(KSP1)                                              DATA5360
      XS(IS)    = 0.5*(XIS1(IS)+XIS2(IS))                               DATA5370
      AO(IS)    = 0.5*(RS(KS)+RS(KSP1))                                 DATA5380
      AOP(IS)   = (RS(KSP1)-RS(KS)) / (XIS2(IS)-XIS1(IS))               DATA5390
  720 CONTINUE                                                          DATA5400
  730 CONTINUE                                                          DATA5410
      IXXTO = IXX                                                       DATA5420
      IF (IXXTO.EQ.0)  GO TO  750                                       DATA5430
      WRITE  (6,130)                                                    DATA5440
      DO 740  I=1,IXXTO                                                 DATA5450
      WRITE  (6,160)  NUBODY(I),NPANEL(I),DETA(I),DZET(I)               DATA5460
  740 CONTINUE                                                          DATA5470
  750 CONTINUE                                                          DATA5480
      IL     = I2                                                       DATA5490
      JL     = J                                                        DATA5500
      ISL    = IS2                                                      DATA5510
C                                                                       DATA5520
C  **   WRITE RECEIVING POINT ARRAYS                                    DATA5530
C                                                                       DATA5540
      WRITE (6,310)                                                     DATA5550
      WRITE (6,70)  (X(I),     I=1,IL)                                  DATA5560
      WRITE (6,70)  (YS(J),    J=1,JL)                                  DATA5570
      WRITE (6,70)  (ZS(J),    J=1,JL)                                  DATA5580
      WRITE (6,70)  (GMAR(J),  J=1,JL)                                  DATA5590
      IF (ISL.EQ.0)  GO TO  760                                         DATA5600
C                                                                       DATA5610
C  **   WRITE SLENDER BODY SENDING ELEMENT ARRAYS                       DATA5620
C                                                                       DATA5630
      WRITE (6,320)                                                     DATA5640
      WRITE (6,70)  (XIS1(I), I=1,ISL)                                  DATA5650
      WRITE (6,70)  (XIS2(I), I=1,ISL)                                  DATA5660
      WRITE (6,70)  (AO(I),   I=1,ISL)                                  DATA5670
      WRITE (6,70)  (AOP(I),  I=1,ISL)                                  DATA5680
  760 CONTINUE                                                          DATA5690
      RETURN                                                            DATA5700
      END                                                               DATA5710
```

```
      SUBROUTINE GEND(NPRINT,NTAPE,WORK)
C     ***   GENERATES THE INFLUENCE COEFFICIENT MATRIX  DT    USING THE
C           FOLLOWING FOUR SUBROUTINES  --  DPPS, DPZY, DZPY,  AND  DYPZ
      COMMON      NP,NTP,NB,NBZ,NBY,NTZ,NTY,FMACH,REFA,REFS,REFC,ND,NE,
     1            NBEA(10,2),NSHEA(10),NCARAY(100),NSARAY(100),NBARAY(100)
     2            ,ZB(10),YB(10),XLE(10),XTE(10),ARB(10),AVR(10),AD(200),
     3            ADP(200),XIS1(200),XIS2(200),XIJ(100),CS(210),YS(210),
     4            ZS(210),EE(200),SG(200),CG(200),X(500),XIC(500),KR,XA,
     5            DELX(500),XLAM(500),RIA(100),THIA(100),TH2A(100),NFL(10)
     6            ,IFLA(30,2),NT12(10,2),NAS(100),NASB(200),YIN(100),
     7            ZIN(100),DETA(50),DZET(50),NBODY(50),NPANEL(50),NTO
     8            ,NTZS,NTYS
      DIMENSION   NTAPE(20)
      COMPLEX     DT(500),DPZ(500),DPY(500)
      DIMENSION   WORK(10000)
      REAL        KR
      EQUIVALENCE  (DT(1),DPZ(1))
   10 FORMAT   (1H0,15I5)
   20 FORMAT   (1H0//30X,26H-DT- MATRIX    **    ROW NO., I4,6H  WITH,I4,
     1          10H  ELEMENTS/)
   30 FORMAT   (1H0,10X,6F18.8)
      ITP1 = NTAPE(1)
      ITP2 = NTAPE(2)
      ITP3 = NTAPE(3)
      ITP4 = NTAPE(4)
      ITP8 = NTAPE(8)
      REWIND  ITP1
      REWIND  ITP2
      REWIND  ITP3
      REWIND  ITP4
      REWIND  ITP5
      EPS  = 0.00001
      PI   = 3.1415926
      FLND = FLOAT(ND)
      FLNE = FLOAT(NE)
      BETA = SQRT(1.0-FMACH**2)
      FL   = REFC
      IBV  = 0
      NBOX = NTP
      LBO  = 1
      LSO  = 1
      JBO  = 1
      KB   = 0
      KT   = 0
      DP   = (0.0,0.0)
      DO 40  I=1,500
      DT(I) = (0.0,0.0)
   40 CONTINUE
      I1   = 1
      I2   = NTP
      J1   = 1
      J2   = NTP

C   DPP-LOOP
      K    = 1
C   K IS THE PANEL NUMBER ASSOCIATED WITH RECEIVING POINT  I
      KS   = 1
C   KS IS THE STRIP NUMBER ASSOCIATED WITH RECEIVING POINT  I
      NBXR = NCARAY(K)
      DO 60  I=I1,I2
      SGR  = SG(KS)
      CGR  = CG(KS)
      CALL        DPPS(K,KS,I,J1,J2,SGR,CGR,REFC,FMACH,YS,ZS,NBARAY,
     1            NCARAY,DT,WORK)
      WRITE  (ITP1) (DT(J),  J=J1,J2)
      IF (I.EQ.I2)  GO TO  60
      IF (I.EQ.NBARAY(K))  K=K+1
      IF (I.EQ.NBXR)  GO TO  50
      GO TO  60
   50 CONTINUE
      KS   = KS+1
      NBXR = NBXR+NCARAY(K)
   60 CONTINUE
      NSTRIP = KS
      NZYSV= 0
      DO  70  J=J1,J2
      DT(J)= (0.0,0.0)
   70 CONTINUE
      IF (NB.EQ.0)  GO TO  320
      IF (NTZ.EQ.0)  GO TO  180
      I1   = I2+1
      I2   = I2+NTZ

C   DPZ-LOOP     **    ALSO USED FOR GENERATING THE DPY-MATRIX  --  SEE
C   COMMENT IN  DPY-LOOP  BELOW
   80 CONTINUE
      KB   = KB+1
C   KB  IS THE BODY NUMBER ASSOCIATED WITH RECEIVING POINT  I
      IZ   = 0
      KT   = KT+1
C   KT  IS THE INDEX OF THE ARRAY OF FIRST-AND-LAST-ELEMENTS FOR THETA-1
C   FOR ALL BODIES  (ARRAY NAME IS  IFLA(100,2))
      ICOUNT = 1
      IFL    = NFL(KB)
      NZYKB= NBEA(KB,2)
      IFIRST = IFLA(KT,1)
      ILAST  = IFLA(KT,2)
      DO  170  I=I1,I2
      DO  90  J=J1,J2
      DPZ(J) = (0.0,0.0)
      DPY(J) = (0.0,0.0)
   90 CONTINUE
      CALL        DPZY(   KB,IZ,I,J1,J2,IFIRST,ILAST,REFC,FMACH,YB,ZB,
     1            AVR,ARB,TH1A,TH2A,NT12,NBARAY,NCARAY,NZYKB,DPZ,DPY,WORK)
      GO TO  (100,100,110),  NZYKB
  100 CONTINUE
      WRITE  (ITP1) (DPZ(J),  J=J1,J2)
      IF (NZYKB.EQ.1)  GO TO 120
  110 CONTINUE
      WRITE  (ITP4) (DPY(J),  J=J1,J2)
  120 CONTINUE
      IF (IZ.EQ.NBEA(KB,1))  GO TO 130
      IF (IZ.EQ.ILAST.AND.ICOUNT.LT.IFL)  GO TO 160
      GO TO  170
  130 CONTINUE
      IZ      = 0
      IF (NZYSV.LE.1.AND.NZYKB.GE.2)  GO TO  140
      GO TO  150
  140 CONTINUE
      LBO  = KB
      LSO  = NSTRIP+LBO
      JBO  = I-NBEA(KB,1)-NBOX+1
  150 CONTINUE
      NZYSV = NZYKB
      KB      = KB+1
      ICOUNT = 0
      IFL     = NFL(KB)
      NZYKB= NBEA(KB,2)
  160 CONTINUE
      KT      = KT+1
      ICOUNT = ICOUNT+1
      IFIRST = IFLA(KT,1)
      ILAST  = IFLA(KT,2)
  170 CONTINUE
  180 CONTINUE
      IF (I2.EQ.NT))  GO TO  190

C   DPY-LOOP     **    THIS LOOP IS REDUCED TO SETTING THE CORRECT INDICES
C   AND USING THE  DPZ-LOOP  ABOVE
      I1   = I2+1
      I2   = NTO
      GO TO 80
  190 CONTINUE
      I1   = 1
      I2   = NTP
      IF (NTZ.EQ.0)  GO TO  250

C   DZP-LOOP
      K    = 1
C   K  IS THE PANEL NUMBER ASSOCIATED WITH RECEIVING POINT  I
      KS   = 1
C   KS  IS THE STRIP NUMBER ASSOCIATED WITH RECEIVING POINT  I
      NBXR = NCARAY(K)
      KB   = 0
C   HERE  KB=0  SERVES AS A FLAG INDICATING THAT THE RECEIVING POINT   I
C   IS ON A PANEL AND NOT  ON A BODY
      J1   = J2+1
      J2   = J2+NTZ
      DO  210  I=I1,I2
```

```
      LS    = NSTRIP+1
      SGR   = SG(KS)
      CGR   = CG(KS)
      CALL        DZPY(KB,KS,LS,IZ,I,J1,J2,NYFLAG,FLND,FLNE,SGR,CGR,REFC,
     1             FMACH,KR,ARB,NBEA,DT)
      WRITE  (ITP2) (DT(J),   J=J1,J2)
      IF (I.EQ.I2)  GO TO  210
      IF (I.EQ.NBARAY(K))  K =K +1
      IF (I.EQ.NBXR)  GO TO  200
      GO TO  210
  200 CONTINUE
      KS    = KS+1
      NBXR = NBXR+NCARAY(K)
  210 CONTINUE

      NYFLAG = 0
C   DZZ-LOOP    **    ALSO USED FOR GENERATING THE   DZY  MATRIX  --  SEE
C   COMMENT IN  DZY-LOOP  BELOW
      KB    = 1
C   KB  IS THE BODY NUMBER ASSOCIATED WITH RECEIVING POINT  I
      KS    = NSTRIP+1
      IZ    = 0
      I1    = I2+1
      I2    = I2+NTZ
      SGR   = 0.0
      CGR   = 1.0
  220 CONTINUE
      LS    = NSTRIP+1
      LSX   = LS
      DO  240  I=I1,I2
      LS    = LSX
      IZ    = IZ+1
C   KS IS THE INDEX OF THE Y  AND  Z  COORDINATES OF RECEIVING POINT I
C   IN THE  DZZ-LOOP  KS RUNS FROM  (NSTRIP+1)  THROUGH  (NSTRIP+NBZ)
C   IN THE  DZY-LOOP  KS  RUNS FROM  (NSTRIP+NB-NBY+1) THROUGH  NSTRIP+NB
      CALL        DZPY(KB,KS,LS,IZ,I,J1,J2,NYFLAG,FLND,FLNE,SGR,CGR,REFC,
     1             FMACH,KR,ARB,NBEA,DT)
      WRITE  (ITP2) (DT(J),   J=J1,J2)
      IF (IZ.EQ.NBEA(KB,1))  GO TO  230
      GO TO  240
  230 CONTINUE
      IZ    = 0
      KB    = KB+1
      KS    = KS+1
  240 CONTINUE
      IF (NTY.EQ.0)  GO TO 320
      IF (NYFLAG.NE.0)  GO TO  250
C   DZY-LOOP    **    THIS LOOP IS REDUCED TO SETTING THE CORRECT INDICES
C   AND USING THE  DZZ-LOOP  ABOVE
      I1    = NTO-NTY+1
      I2    = NTO
      NYFLAG = 1
      KB    = LBO
      KS    = LSO
      SGR   =-1.0
      CGR   = 0.0
      GO TO  220
  250 CONTINUE
      IF (NTY.EQ.0)   GO TO 320
      I1    = 1
      I2    = NTP
C   DYP-LOOP
      K     = 1
      KS    = 1
      KB    = 0
      NBXR = NCARAY(K)
      J1    = NTO-NTY+1
      J2    = NTO
      SGR   = SG(KS)
      CGR   = CG(KS)
      DO  270  I=I1,I2
      CALL        DYPZ(KB,KS,LS,IZ,I,J1,J2,NYFLAG,FLND,FLNE,SGR,CGR,REFC,
     1             FMACH,KR,ARB,NBEA,LBO,LSO,JB3,DT)
      WRITE  (ITP3) (DT(J),   J=J1,J2)
      IF (I.EQ.NBARAY(K))  K=K+1
      IF (I.EQ.NBXR)  GO TO  260
      GO TO  270
  260 CONTINUE
      KS    = KS+1
      NBXR = NBXR+NCARAY(K)
      SGR   = SG(KS)
      CGR   = CG(KS)
  270 CONTINUE
      NYFLAG = 0
      IZ    = 0
      IF (NTZ.EQ.0)   GO TO  310
C   DYZ-LOOP    **    ALSO USED FOR GENERATING THE   DYY  MATRIX  --  SEE
C   COMMENT IN  DYY-LOOP  BELOW
      I1    = I2+1
      I2    = I2+NTZ
      KS    = NSTRIP+1
      KB    = 1
      SGR   = 0.0
      CGR   = 1.0
  280 CONTINUE
      DO  300  I=I1,I2
      LS    = LSO
      IZ    = IZ+1
      CALL        DYPZ(KB,KS,LS,IZ,I,J1,J2,NYFLAG,FLND,FLNE,SGR,CGR,REFC,
     1             FMACH,KS,ARB,NBEA,LBO,LSO,JB3,DT)
      WRITE  (ITP3) (DT(J),   J=J1,J2)
      IF (IZ.EQ.NBEA(KB,1))  GO TO  290
      GO TO  300
  290 CONTINUE
      IZ    = 0
      KB    = KB+1
      KS    = KS+1
  300 CONTINUE
  310 CONTINUE
      IF (NYFLAG.NE.0)   GO TO 320
C   DYY-LOOP    **    THIS LOOP IS REDUCED TO SETTING THE CORRECT INDICES
C   AND USING THE  DYZ-LOOP  ABOVE
      I1    = NTO-NTY+1
      I2    = NTO
      NYFLAG = 1
      KB    = LBO
      KS    = LSO
      SGR   =-1.0
      CGR   = 0.0
      GO TO  280
  320 CONTINUE
      REWIND  ITP1
      REWIND  ITP2
      REWIND  ITP3
      REWIND  ITP4
      REWIND  ITP8
      I1    = 1
      I2    = NTP+NTZ
      NYFLAG = 0
      ITAPE = 1
  330 CONTINUE
      DO 360  I=I1,I2
      J1    = 1
      J2    = NTP
      READ  (ITAPE) (DT(J),   J=J1,J2)
      IF (NTZ.EQ.0)   GO TO 340
      J1    = J2+1
      J2    = J2+NTZ
      READ   (ITP2) (DT(J),   J=J1,J2)
  340 CONTINUE
      IF (NTY.EQ.0)   GO TO 350
      J1    = J2+1
      J2    = J2+NTY
      READ   (ITP3) (DT(J),   J=J1,J2)
  350 CONTINUE
      WRITE  (ITP8) (DT(J), J=1,J2)
      IF (NPRINT.EQ.0)   GO TO 360
      WRITE  (6,20)   I,J2
      WRITE  (6,30)   (DT(J),   J=1 ,J2)
  360 CONTINUE
      IF (NTY.EQ.0)   GO TO 370
      IF (NYFLAG.NE.0)   GO TO 370
      NYFLAG = 1
      I1    = I2+1
      I2    = I2+NTY
      ITAPE  = 4
      GO TO 330
  370 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE DPPS(K,KS,I,J1,J2,SGR,CGR,REFC,FMACH,YS,ZS,NBARAY,      DPPS0010
     1              NCARAY,DT,WORK)                                       DPPS0020
C   ***   GENERATES ROWS OF THE  DPP  SUBMATRIX USING                     DPPS0030
C           SUBROUTINE  SUBP                                              DPPS0040
      REAL        KP                                                      DPPS0050
      DIMENSION  YS(200),ZS(200),NBARAY(100),NCARAY(100)                  DPPS0060
      COMPLEX     SUM,DT(500)                                             DPPS0070
      DIMENSION  WORK(10000)                                              DPPS0080
      BETA = SQRT(1.0-FMACH**2)                                           DPPS0090
      FL   = REFC                                                         DPPS0100
      L    = 1                                                            DPPS0110
C  L IS THE PANEL NUMBER ASSOCIATED WITH SENDING   POINT  J               DPPS0120
      LS   = 1                                                            DPPS0130
C  LS IS THE STRIP NUMBER ASSOCIATED WITH SENDING   POINT  J              DPPS0140
      IO   = 0                                                            DPPS0150
      NBXS = NBARAY(L)                                                    DPPS0160
      NCI  = NCARAY(L)                                                    DPPS0170
      NCPNB= NCI                                                          DPPS0180
      NBCUM= NCI                                                          DPPS0190
      YREC = YS(KS)                                                       DPPS0200
      ZREC = ZS(KS)                                                       DPPS0210
      DO   20  J=J1,J2                                                    DPPS0220
      IO   = IO+1                                                         DPPS0230
      IR   = J                                                            DPPS0240
      CALL         SUBP(I,L,LS,J,IO,IR,NBXS,NCPNB,SGR,CGR,YREC,ZREC,SUM,  DPPS0250
     1              WORK)                                                 DPPS0260
      DT(J)= SUM                                                          DPPS0270
      IF (J.EQ.J2)  GO TO   20                                            DPPS0280
      IF (IO.EQ.NCI)  IO=0                                                DPPS0290
      IF (J.LT.NBXS)   GO TO  10                                          DPPS0300
      L    = L+1                                                          DPPS0310
      NCI  = NCARAY(L)                                                    DPPS0320
      NCPNB= NBXS + NCI                                                   DPPS0330
      NBXS = NBARAY(L)                                                    DPPS0340
   10 CONTINUE                                                            DPPS0350
      IF (J.LT.NBCUM)  GO TO  20                                          DPPS0360
      LS   = LS+1                                                         DPPS0370
      NBCUM= NBCUM+NCI                                                    DPPS0380
   20 CONTINUE                                                            DPPS0390

      RETURN                                                              DPPS0400
      END                                                                 DPPS0410
```

```
      SUBROUTINE DPZY(   KB,IZ,I,J1,J2,IFIRST,ILAST,REFC,FMACH,YB,ZB,     DPZY0010
     1              AVR,ARB,TH1A,TH2A,NT12,NBARAY,NCARAY,NZYKB,DPZ,DPY,WORK)DPZY0020
C   ***   GENERATES ROWS OF THE SUBMATRICES  DPZ  AND DPY  USING          DPZY0030
C           SUBROUTINE  SUBP                                              DPZY0040
      COMPLEX     SUM,DPZ(500),DPY(500)                                   DPZY0050
      DIMENSION  YB(10),ZB(10),AVR(10),ARB(10),TH1A(100),TH2A(100),       DPZY0060
     1              NT12(10,2),NBARAY(100),NCARAY(100)                    DPZY0070
      DIMENSION  WORK(10000)                                              DPZY0080
   10 FORMAT  (1H0,15I5)                                                  DPZY0090
   20 FORMAT  (1H0,10X,6E18.8)                                            DPZY0100
      PI   = 3.1415926                                                    DPZY0110
      BETA = SQRT(1.0-FMACH**2)                                           DPZY0120
      FL   = REFC                                                         DPZY0130
      NBV  = 0                                                            DPZY0140
      IX1    = 1                                                          DPZY0150
      I2   = IZ+1                                                         DPZY0160
C  IZ  IS THE BODY-ELEMENT NUMBER FOR BODY  KB  --  IZ RUNS FROM  1       DPZY0170
C  THROUGH  NBE-SUB-KB                                                    DPZY0180
      IX2    = NT12(KB,2)                                                 DPZY0190
      IF (IZ.GE.IFIRST.AND.IZ.LE.ILAST)  IX2=NT12(KB,1)                   DPZY0200
      DO  100  IX=IX1,IX2                                                 DPZY0210
      DPUR = (0.0,0.0)                                                    DPZY0220
      DPUL = (0.0,0.0)                                                    DPZY0230
      DPLR = (0.0,0.0)                                                    DPZY0240
      DPLL = (0.0,0.0)                                                    DPZY0250
      L    = 1                                                            DPZY0260
C  L IS THE PANEL NUMBER ASSOCIATED WITH SENDING   POINT  J               DPZY0270
      LS   = 1                                                            DPZY0280
C  LS IS THE STRIP NUMBER ASSOCIATED WITH SENDING   POINT  J              DPZY0290
      IO   = 0                                                            DPZY0300
      NBXS = NBARAY(L)                                                    DPZY0310
      NCI  = NCARAY(L)                                                    DPZY0320
      NCPNB= NCI                                                          DPZY0330
      NBCUM= NCI                                                          DPZY0340
      IXP1 = IX+1                                                         DPZY0350
      IF (IXP1.GT.IX2)  IXP1=IX1                                          DPZY0360
      IXM1 = IX-1                                                         DPZY0370
      IF (IXM1.EQ.0)   IXM1=IX2                                           DPZY0380
      IF (IZ.GE.IFIRST.AND.IZ.LE.ILAST)  GO TO  30                        DPZY0390

      THETA= TH2A(IX)                                                     DPZY0400
      THP1 = TH2A(IXP1)                                                   DPZY0410
      THM1 = TH2A(IXM1)                                                   DPZY0420
      GO TO  40                                                           DPZY0430
   30 CONTINUE                                                            DPZY0440
      THETA = TH1A(IX)                                                    DPZY0450
      THP1 = TH1A(IXP1)                                                   DPZY0460
      THM1 = TH1A(IXM1)                                                   DPZY0470
   40 CONTINUE                                                            DPZY0480
      IF (IX.EQ.IX1)  THM1=THM1-2.0*PI                                    DPZY0490
      IF (IX.EQ.IX2)  THP1=THP1+2.0*PI                                    DPZY0500
      DELTH= 0.5*(THP1 - THM1)                                            DPZY0510
      YREC = YB(KB)+AVR(KB)*COS(THETA)                                    DPZY0520
      ZREC = ZB(KB)+AVR(KB)*ARB(KB)*SIN(THETA)                            DPZY0530
      RHO  = SQRT(1.0+(ARB(KB)**2 - 1.0) * (COS(THETA))**2)               DPZY0540
      SGR  = -ARB(KB)*COS(THETA)/RHO                                      DPZY0550
      CGR  = SIN(THETA)/RHO                                               DPZY0560
      SMULT= SIN(THETA) * RHO / PI                                        DPZY0570
      CMULT= COS(THETA) * RHO / PI                                        DPZY0580
      DO  90  J=J1,J2                                                     DPZY0590
      IO   = IO+1                                                         DPZY0600
      IR   = J                                                            DPZY0610
      CALL         SUBP(I,L,LS,J,IO,IR,NBXS,NCPNB,SGR,CGR,YREC,ZREC,SUM,  DPZY0620
     1              WORK)                                                 DPZY0630
      GO TO  (50,50,60),  NZYKB                                           DPZY0640
   50 CONTINUE                                                            DPZY0650
      DPZ(J) = DPZ(J) + SUM * SMULT * DELTH                               DPZY0660
      IF (NZYKB.EQ.1)  GO TO 70                                           DPZY0670
   60 CONTINUE                                                            DPZY0680
      DPY(J) = DPY(J) + SUM * CMULT * DELTH                               DPZY0690
   70 CONTINUE                                                            DPZY0700
      IF (J.EQ.J2)  GO TO  90                                             DPZY0710
      IF (IO.EQ.NCI)  IO=0                                                DPZY0720
      IF (J.LT.NBXS)   GO TO  80                                          DPZY0730
      L      = L+1                                                        DPZY0740
      NCI  = NCARAY(L)                                                    DPZY0750
      NCPNB= NBXS + NCI                                                   DPZY0760
      NBXS = NBARAY(L)                                                    DPZY0770
   80 CONTINUE                                                            DPZY0780
      IF (J.LT.NBCUM)  GO TO  90                                          DPZY0790

      LS   = LS+1                                                         DPZY0800
      NBCUM= NBCUM+NCI                                                    DPZY0810
   90 CONTINUE                                                            DPZY0820
  100 CONTINUE                                                            DPZY0830
      RETURN                                                              DPZY0840
      END                                                                 DPZY0850
```

```
      SUBROUTINE DYPZ(KB,KS,LS,IZ,I,J1,J2,NYFLAG,FLND,FLNF,SGR,CGR,REFC,DYPZ0010
     1              FMACH,KR,ARB,NBEA,LBO,LSO,JBO,DT)                     DYPZ0020
C   ***   GENERATES ROWS OF THE SUBMATRICES  DYP, DYZ  AND DYY            DYPZ0030
C           USING  SUBROUTINE  SUBB                                       DYPZ0040
      REAL        KR,M                                                    DYPZ0050
      COMPLEX     SUM,DT(500)                                             DYPZ0060
      DIMENSION  ARB(10),NBEA(10,2)                                       DYPZ0070
```

```
10 FORMAT  (1H0,6E20.8)                                                 DYPZ0080
   NDY  = 1                                                             DYPZ0090
   NYFL = NYFLAG                                                        DYPZ0100
   IND  = 0                                                             DYPZ0110
   PI   = 3.1415926                                                     DYPZ0120
   EPS  = 0.00001                                                       DYPZ0130
   BETA = SQRT(1.0-FMACH**2)                                            DYPZ0140
   FL   = REFC                                                          DYPZ0150
   M    = FMACH                                                         DYPZ0160
   NBV  = 0                                                             DYPZ0170
   JZ   = 0                                                             DYPZ0180
   LB   = LB0                                                           DYPZ0190
C  LB  IS THE BODY NUMBER ASSOCIATED WITH SENDING POINT  J              DYPZ0200
   SGS  = -1.0                                                          DYPZ0210
   CGS  = 0.0                                                           DYPZ0220
   LS   = LS0                                                           DYPZ0230
C  LS IS THE INDEX OF THE  Y  AND  Z  COORDINATES OF SENDING POINT  J --DYPZ0240
C  LS RUNS FROM NSTRIP+NB-NBY+1  THROUGH  NSTRIP+NB                     DYPZ0250
   JB   = JB0-1                                                         DYPZ0260
   NZYLB= NBEA(LB,2)                                                    DYPZ0270
   AR   = ARB(LB)                                                       DYPZ0280
   SL   = 0.0                                                           DYPZ0290
   CL   = 1.0                                                           DYPZ0300
   TL   = 0.0                                                           DYPZ0310
   DO  30  J=J1,J2                                                      DYPZ0320
   JB   = JB+1                                                          DYPZ0330
   JZ   = JZ+1                                                          DYPZ0340
   CALL        SUBB(KB,KS,I,J,JZ,JB,LB,LS,NDY,NYFL,FLND,FLNE,PI,EPS,    DYPZ0350
  1            SGR,CGR,SGS,CGS,AR,SL,CL,TL,FL,BETA,SUM)                 DYPZ0360
   DT(J)= SUM                                                           DYPZ0370
   IF  (JZ.EQ.NBEA(LB,1))  GO TO  20                                    DYPZ0380
   GO TO 30                                                             DYPZ0390

20 CONTINUE                                                             DYPZ0400
   JZ   = 0                                                             DYPZ0410
   LB   = LB+1                                                          DYPZ0420
   LS   = LS+1                                                          DYPZ0430
   AR   = ARB(LB)                                                       DYPZ0440
   NZYLB= NBEA(LB,2)                                                    DYPZ0450
30 CONTINUE                                                             DYPZ0460
   RETURN                                                               DYPZ0470
   END                                                                  DYPZ0480
```

```
   SUBROUTINE DZPY(KB,KS,LS,IZ,I,J1,J2,NYFLAG,FLND,FLNE,SGR,CGR,REFC,DZPY0010
  1            FMACH,KR,ARB,NBEA,DT)                                    DZPY0020
C  ***  GENERATES ROWS OF THE SUBMATRICES  DZP, DZZ  AND  DZY           DZPY0030
C       USING  SUBROUTINE  SUBB                                         DZPY0040
   REAL        KR,M                                                     DZPY0050
   DIMENSION   ARB(10),NBEA(10,2)                                       DZPY0060
   COMPLEX     SUM,DT(500)                                              DZPY0070
10 FORMAT  (1H0,6E20.8)                                                 DZPY0080
   NDY  = 0                                                             DZPY0090
   NYFL = NYFLAG                                                        DZPY0100
   IND  = 0                                                             DZPY0110
   NZYKB  = NBEA(KB,2)                                                  DZPY0120
   PI   = 3.1415926                                                     DZPY0130
   EPS  = 0.00001                                                       DZPY0140
   BETA = SQRT(1.0-FMACH**2)                                            DZPY0150
   FL   = REFC                                                          DZPY0160
   M    = FMACH                                                         DZPY0170
   NBV  = 0                                                             DZPY0180
   JZ   = 0                                                             DZPY0190
   LB   = 1                                                             DZPY0200
   JB   = 0                                                             DZPY0210
   SGS  = 0.0                                                           DZPY0220
   CGS  = 1.0                                                           DZPY0230
   NZYLB= NBEA(LB,2)                                                    DZPY0240
   AR   = ARB(LB)                                                       DZPY0250
   SL   = 0.0                                                           DZPY0260
   CL   = 1.0                                                           DZPY0270
   TL   = 0.0                                                           DZPY0280
```

```
C  LS IS THE INDEX OF THE  Y  AND  Z  COORDINATES OF SENDING POINT  J --DZPY0290
C  LS RUNS FROM NSTRIP+1  THROUGH  NSTRIP+NBZ                           DZPY0300
   DO  30  J=J1,J2                                                      DZPY0310
   JB   = JB+1                                                          DZPY0320
C  JB  IS THE BODY-ELEMENT NUMBER IN BODY  LB  --  JB  RUNS FROM  1     DZPY0330
C  THROUGH  NZZ                                                         DZPY0340
   JZ   = JZ+1                                                          DZPY0350
C  JZ  RUN  FROM 1  THROUGH  NBE-SUB-LB                                 DZPY0360
   CALL        SUBB(KB,KS,I,J,JZ,JB,LB,LS,NDY,NYFL,FLND,FLNE,PI,EPS,    DZPY0370
  1            SGR,CGR,SGS,CGS,AR,SL,CL,TL,FL,BETA,SUM)                 DZPY0380
   DT(J)= SUM                                                           DZPY0390

   IF  (JZ.EQ.NBEA(LB,1))  GO TO  20                                    DZPY0400
   GO TO 30                                                             DZPY0410
20 CONTINUE                                                             DZPY0420
   JZ   = 0                                                             DZPY0430
   LB   = LB+1                                                          DZPY0440
   LS   = LS+1                                                          DZPY0450
   AR   = ARB(LB)                                                       DZPY0460
   NZYLB= NBEA(LB,2)                                                    DZPY0470
30 CONTINUE                                                             DZPY0480
   RETURN                                                               DZPY0490
   END                                                                  DZPY0500
```

```
   SUBROUTINE SUBB(KB,KS,I,J,JZ,JB,LB,LS,NDY,NYFL,FLND,FLNE,PI,EPS,     SUBB0010
  1            SGR,CGR,SGS,CGS,AR,SL,CL,TL,FL,BETA,SUM)                 SUBB0020
C  ***  COMPUTES ELEMENTS OF THE SUBMATRICES  DZP, DZZ, DZY, DYP,       SUBB0030
C       DYZ  AND  DYY  USING  SUBROUTINE  DZY                           SUBB0040
   COMMON      NP,NTP,NB,NBZ,NBY,NTZ,NTY,FMACH,REFA,REFS,REFC,ND,NE,    SUBB0050
  1            NBEA(10,2),NSBEA(10),NCARAY(100),NSARAY(100),NBARAY(10)) SUBB0060
  2            ,ZB(10),YB(10),XLE(10),XTE(10),ARB(10),AVR(10),AC(200),  SUBB0070
  3            A0P(200),XIS1(200),XIS2(200),XIJ(100),CS(210),YS(210),   SUBB0080
  4            ZS(210),EE(200),SG(200),CG(200),X(500),XIC(500),KR,XM,   SUBB0090
  5            DELX(500),XLAM(500),RIA(100),TH1A(100),TH2A(100),NFL(10) SUBB0100
  6            ,IFLA(30,2),NT12(10,2),NAS(100),NAS3(200),YIN(100),      SUBB0110
  7            ZIN(100),DETA(50),DZET(50),NOBODY(50),NPANEL(50),NTO     SUBB0120
  8            ,NTZS,NTYS                                               SUBB0130
   COMMON      /KDS/ IND,KD1R,KD1I,KD2R,KD2I                            SUBB0140
   REAL        KD1R,KD1I, KD2R,KD2I                                     SUBB0150
   REAL        KR,M                                                     SUBB0160
   COMPLEX     DPUR,DPUL,DPLR,DPLL,DP,SUM,FIKX                          SUBB0170
10 FORMAT  (1H0,6E20.8)                                                 SUBB0180
   IPRNT  = 0                                                           SUBB0190
   IND  = 0                                                             SUBB0200
   BR   = FL/2.0                                                        SUBB0210
   DPUR = (0.0,0.0)                                                     SUBB0220
   DPUL = (0.0,0.0)                                                     SUBB0230
   DPLR = (0.0,0.0)                                                     SUBB0240
   DPLL = (0.0,0.0)                                                     SUBB0250
   ANOT = RIA(JB)                                                       SUBB0260
   DXS  = DELX(J)                                                       SUBB0270
   ABSYB= ABS(YB(LB))                                                   SUBB0280
   ABSZB= ABS(ZB(LB))                                                   SUBB0290
                                                                        SUBB0300
                                                                        SUBB0310
   IF (KB.EQ.0)  GO TO  20                                              SUBB0320
   TEST1= ABS(YB(LB) -YB(KB))                                           SUBB0330
   TEST2= ABS(ZB(LB) -ZB(KB))                                           SUBB0340
   IF  ((TEST1.GT.EPS. OR .TEST2.GT.EPS)  GO TO  20                     SUBB0350
   SUM  = (0.0,0.0)                                                     SUBB0360
   IF (NDY.NE.NYFL)  GO TO  100                                         SUBB0370
   IF (I.NE.J)   GO TO 100                                              SUBB0380
   D2D  =        1.0 /(2.0*PI*ANOT*ANOT*(1.0+AR))                       SUBB0390
```

```
      IF     (NDY.NE.0)   D2D=D2D/AR
      SUM   = CMPLX(D2D,0.0)
      GO TO  100
   20 CONTINUE
      XX    = X(I)
      Y     = YS(KS)
      Z     = ZS(KS)
      XI1   = X(J) - 0.5*DXS
      XI2   = X(J) + 0.5*DXS
      ETA   = YS(LS)
      ZETA  = ZS(LS)
      A0    = ANOT
      IOZOY= NOY
      IGO   = 1
   30 CONTINUE
      CALL         D2Y   (XX, Y, Z, SGS, CGS, XI1, XI2, ETA, ZETA, AR, A0,
     1        KR    ,   BR    ,   BETA   ,   FMACH ,
     2        IOZOY ,   DZDYR ,   DZDYI  , IPRNT  )
      DP    = CMPLX(DZDYR,DZDYI)
      GO TO  (40,50,70,80),  IGO
   40 CONTINUE
C     UPPER RIGHT-HAND SIDE CONTRIBUTION
      DPUR = DP
      IF (KR.EQ.LB)  GO TO 100
      IF (ABSYR.LE.EPS.AND.ABSZR.LE.EPS)  GO TO  90
      IF (ND.EQ.0 OR.ABSYB.LE.EPS)  GO TO  60
      IGO   = 2
      ETA   = -YS(LS)
      GO TO  30
   50 CONTINUE
C     UPPER LEFT-HAND SIDE CONTRIBUTION
      DPUL = DP
   60 CONTINUE
      IF (NE.EQ.0.OR.ABSZB.LE.EPS)  GO TO  90
      IGO   = 3
      ETA   =  YS(LS)
      ZETA  = -ZS(LS)
      GO TO  30
   70 CONTINUE
C     LOWER RIGHT-HAND SIDE CONTRIBUTION
      DPLR = DP
      IF (ND.EQ.0.OR.ABSYB.LE.EPS)  GO TO  90
      IGO   = 4
      ETA   = -YS(LS)
      ZETA  = -ZS(LS)
      GO TO  30
   80 CONTINUE
C     LOWER  LEFT-HAND SIDE CONTRIBUTION
      DPLL = DP
   90 CONTINUE
      SUM   = DPUR + FLND*DPUL + FLNE*DPLR + FLND*FLNE*DPLL
  100 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SUBP(I,L,LS,J,IO,IR,NBXS,NCPNB,SGR,CGR,YREC,ZREC,SUM,
     1               WORK)
C     ***   COMPUTES ELEMENTS OF THE SUBMATRICES  DPP, DPZ  AND  DPY
C           USING  SUBROUTINES  SNPDF,  INCRO  AND SUBI
      COMMON        NP,NTP,NB,NPZ,NPY,NTZ,NTY,FMACH,REFA,REFS,REFC,ND,NE,
     1             NBEA(10,2),NSBEA(10),NCARAY(100),NSARAY(100),NBARAY(10)
     2             ,ZB(10),YB(10),XLE(10),XTE(10),ARB(10),AVR(10),A0(200),
     3             ACP(200),XIS1(200),XIS2(200),XIJ(100),ZS(210),YS(210),
     4             ZS(210),EE(200),SG(200),CG(200),XI(500),XIC(500),KR,XM,
     5             DELX(500),XLAM(500),PIA(100),THIA(100),TH2A(100),NFL(10)
     6             ,IFLA(20,2),NT12(10,2),FAS(100),MASR(2,0),YIN(100),
     7             ZIN(100),DETX(50),DZET(50),NFBODY(10),NPANEL(50),NTO
     8             ,NTZS,NTYS
      COMPLEX       DPUR,DPUL,DPLR,DPLL,DP,SUM
      DIMENSION     WORK(10000)
      REAL          KR,M
      DATA          NWK1,NWK2,NWK3,NWK4 /    1, 701,1401,2101/
      DATA          NWK5,NWK6,NWK7,NWK8 /2801,3501,4201,4901/
   10 FORMAT   (1H0,15I5)
   20 FORMAT   (1H0,6E20.8)
      PI    = 3.1415926
      EPS   = 0.00001
      M     = FMACH
      BETA  = SQRT(1.0-M*M)
      FL    = REFC
      NBV   = 0
      FLND  = FLOAT(ND)
      FLNE  = FLOAT(NE)
      SGS   = SG(LS)
      CGS   = CG(LS)
      DPUR  = (0.0,0.0)
      DPUL  = (0.0,0.0)
      DPLR  = (0.0,0.0)
      DPLL  = (0.0,0.0)
      DIJ   = 0.0
      DELR  = 0.0
      DELL  = 0.0
      DIJI  = 0.0
      DELRI= 0.0

      DELII= 0.0
      INFL    = 0
      IOUTFL  = 0
      IMGS    = 0
C     UPPER RIGHT SENDING POINT
      IGO   = 1
      LHS   = 0
      NOBLE= 0
      TL    = XLAM(J)
      SQTL  = SQRT(1.0+TL**2)
      SL    = TL / SQTL
      CL    = 1.0/ SQTL
      X0    = X(I)   - XIC(J)
      Y0    = YREC -YS(LS)
      Z0    = ZREC -ZS(LS)
      ES    = EE(LS)
      DXS   = DELX(J)
      AX    = X0
      AY    = Y0
      AZ    = Z0
      CV    = DXS
   30 CONTINUE
      IMG   = 0
      NOBI  = 1
      NAZ   = 0
      CALL        SNPDF(SL,CL,TL,SGS,CGS,SGR,CGR,X0,Y0,Z0,ES,DIJ,BETA,CV)
      IF (KR.LE.EPS)  GO TO  40
      SDELX= DXS
      DELY  = 2.0*ES
      AX1   = AX + ES*TL
      AY1   = AY + ES*CGS
      AZ1   = AZ + ES*SGS
      AX2   = AX - ES*TL
      AY2   = AY - ES*CGS
      AZ2   = AZ - ES*SGS
      CALL        INCRO(AX,AY,AZ,AX1,AY1,AZ1,AX2,AY2,AZ2,SGR,CGR,SGS,CGS,
     1        KR,FL,BETA,SDELX,DELY,DELF,DELI,IO,IF,NBXS,NCPNB,LHS,
     2        NOBLE,IMG,NOBI,IMGS,   WORK(NWK1),WORK(NWK2),WORK(NWK3),
     3        WORK(NWK4),WORK(NWK5),WORK(NWK6),WORK(NWK7),WORK(NWK8))
   40 CONTINUE
```

```
      IF  (NB.EQ.0)  GO TO  120                                         SUBP0800
      NOAS = NAS(L)                                                     SUBP0810
C  ** CHECK FOR ASSOCIATED BODIES                                       SUBP0820
      IF (NOAS.EQ.0)  GO TO  120                                        SUBP0830
      DIJS = DIJ                                                        SUBP0840
      DELRS= DELR                                                       SUBP0850
      DELIS= DELI                                                       SUBP0860
      DIJI = 0.0                                                        SUBP0870
      DELRI= 0.0                                                        SUBP0880
      DELII= 0.0                                                        SUBP0890
      NA1  = NA2+1                                                      SUBP0900
      NA2  = NA2+NOAS                                                   SUBP0910
      IF  ( NA2.GT.NB)  NA2=NB                                          SUBP0920
      IMG  = 1                                                          SUBP0930
C  ** START DO-LOOP FOR THE SUMMATION OF THE WING-IMAGE CONTRIBUTIONS   SUBP0940
C  OVER  RANGE(P)                                                       SUBP0950
      DO 110  NA=NA1,NA2                                                SUBP0960
      NOB  = NASB(NA)                                                   SUBP0970
C  ** NOB IS THE SEQUENCE NUMBER OF THE CURRENT BODY ASSOCIATED WITH    SUBP0980
C  PANEL  L  IN WHICH THE SENDING POINT  J  LIES                        SUBP0990
      NOBI = NOB                                                        SUBP1000
      DA   = AVR(NOB)                                                   SUBP1010
      DZB  = ZB(NOB)                                                    SUBP1020
      DYB  = YB(NOB)                                                    SUBP1030
      DAR  = ARB(NOB)                                                   SUBP1040
      DXLE = XLE(NOB)                                                   SUBP1050
      DXTE = XTE(NOB)                                                   SUBP1060
      GO TO  (50,60,70,80),  IGO                                        SUBP1070
   50 CONTINUE                                                          SUBP1080
      DETA = YS(LS)                                                     SUBP1090
      DZETA= ZS(LS)                                                     SUBP1100
      GO TO  90                                                         SUBP1110
   60 CONTINUE                                                          SUBP1120
      DETA =-YS(LS)                                                     SUBP1130
      DZETA= ZS(LS)                                                     SUBP1140
      GO TO  90                                                         SUBP1150
   70 CONTINUE                                                          SUBP1160
      DETA = YS(LS)                                                     SUBP1170
      DZETA=-ZS(LS)                                                     SUBP1180
      GO TO  90                                                         SUBP1190

   80 CONTINUE                                                          SUBP1200
      DETA =-YS(LS)                                                     SUBP1210
      DZETA=-ZS(LS)                                                     SUBP1220
   90 CONTINUE                                                          SUBP1230
      DCGAM= CGS                                                        SUBP1240
      DSGAM= SGS                                                        SUBP1250
      DFE  = FS                                                         SUBP1260
      DXI  = XIC(J)                                                     SUBP1270
      IF  (DXI.LT.DXLE.OR.DXI.GT.DXTE)  GO TO 110                       SUBP1280
      CALL       SUPI(DA,DZB,DYB,DAR,DETA,DZETA,DCGAM,DSGAM,DFE,DXI,TL, SUBP1290
     1 DETAI,DZETAI,DCGAMI,DSGAMI,DEFI,DTLAMI,DMUY,DMUZ,INFL,IOUTFL)    SUBP1300
      DIJ  = 0.0                                                        SUBP1310
      IF  (INFL.NE.0.OR.IOUTFL.EQ.0)  GO TO 100                         SUBP1320
      DTL    = DTLAMI                                                   SUBP1330
      DSQRTL = SQRT(1.0+DTL**2)                                         SUBP1340
      DSL    = DTL/DSQRTL                                               SUBP1350
      DCL    = 1.0/DSQRTL                                               SUBP1360
      X0I  = X0                                                         SUBP1370
      Y0I  = YREC - DETAI                                               SUBP1380
      Z0I  = ZREC - DZETAI                                              SUBP1390
      CALL       SNPDF(DSL,DCL,DTL,DSGAMI,DCGAMI,SGR,CGR,X0I,Y0I,Z0I,   SUBP1400
     1     DEFI,DIJ,BETA,CV)                                            SUBP1410
      DIJI = DIJI+DIJ                                                   SUBP1420
      IF  (KR.LE.EPS)  GO TO 100                                        SUBP1430
      DELR = 0.0                                                        SUBP1440
      DELI = 0.0                                                        SUBP1450
      AYI  = Y0I                                                        SUBP1460
      AZI  = Z0I                                                        SUBP1470
      AY1I = AYI - DEFI*DCGAMI                                          SUBP1480
      AZ1I = AZI - DEFI*DSGAMI                                          SUBP1490
      AY2I = AYI + DEFI*DCGAMI                                          SUBP1500
      AZ2I = AZI + DEFI*DSGAMI                                          SUBP1510
      DEFI2= 2.0*DEFI                                                   SUBP1520
      CALL       INCRO(AX,AYI,AZI,AX1,AY1I,AZ1I,AX2,AY2I,AZ2I,SGR,CGR,  SUBP1530
     1    DSGAMI,DCGAMI,KR,FL,BETA,SDELX,DEFI2,DELR,DELI,IO,IR,NBXS,    SUBP1540
     2    NCPNB,LHS,NDBLF,IMG,NOBI,IMGS,WORK(1),WORK(NWK2),WORK(NWK3),  SUBP1550
     3    WORK(NWK4),WORK(NWK5),WORK(NWK6),WORK(NWK7),WORK(NWK8))       SUBP1560
      DELRI= DELRI+DELR                                                 SUBP1570
      DELII= DELII+DELI                                                 SUBP1580
      IF  (IO . EQ . NCARAY(L))  IMGS=1                                 SUBP1590
      GO TO  110                                                        SUBP1600
  100 CONTINUE                                                          SUBP1610
      DELRI= 0.0                                                        SUBP1620
      DELII= 0.0                                                        SUBP1630
  110 CONTINUE                                                          SUBP1640
      DIJ  = DIJS                                                       SUBP1650
      DELR = DELRS                                                      SUBP1660
      DELI = DELIS                                                      SUBP1670
  120 CONTINUE                                                          SUBP1680
      DP   = CMPLX(((DIJ +DIJI)-(DELR +DELRI)), (-DELI -DELII))         SUBP1690
  130 CONTINUE                                                          SUBP1700
      GO TO  (140,150,170,180),  IGO                                    SUBP1710
  140 CONTINUE                                                          SUBP1720
      DPUR = DP                                                         SUBP1730
      IF (ND.EQ.0.OR.(ABS(YS(LS))).LE.0.001)  GO TO  160                SUBP1740
C  UPPER LEFT  SENDING POINT                                            SUBP1750
      IGO  = 2                                                          SUBP1760
      LHS  = 1                                                          SUBP1770
      SGS  = -SGS                                                       SUBP1780
      TL   = -TL                                                        SUBP1790
      SL   = -SL                                                        SUBP1800
      YO   = YREC + YS(LS)                                              SUBP1810
      AY   = YO                                                         SUBP1820
      GO TO   30                                                        SUBP1830
  150 CONTINUE                                                          SUBP1840
      DPUL = DP                                                         SUBP1850
  160 CONTINUE                                                          SUBP1860
      IF (NE.EQ.0.OR.(ABS(ZS(LS))).LE.0.001)  GO TO  190                SUBP1870
C  LOWER RIGHT SENDING POINT                                            SUBP1880
      IGO  = 3                                                          SUBP1890
      LHS  = 0                                                          SUBP1900
      NDBLF= 1                                                          SUBP1910
      TL   =  XLAM(J)                                                   SUBP1920
      SL   =  TL/(SQRT(1.0+TL*TL))                                      SUBP1930
      YO   = YREC -YS(LS)                                               SUBP1940
      ZO   = ZREC + ZS(LS)                                              SUBP195
      AY   = YO                                                         SUBP1960
      AZ   = ZO                                                         SUBP1970
      SGS  =-SG(LS)                                                     SUBP1980
      GO TO   30                                                        SUBP1990

  170 CONTINUE                                                          SUBP2000
      DPLR = DP                                                         SUBP2010
      IF (ND.EQ.0.OR.(ABS(YS(LS))).LE.0.001)  GO TO  190                SUBP2020
C  LOWER LEFT  SENDING POINT                                            SUBP2030
      IGO  = 4                                                          SUBP2040
      LHS  = 1                                                          SUBP2050
      SGS  = SG(LS)                                                     SUBP2060
      TL   = -XLAM(J)                                                   SUBP2070
      SL   =  TL/(SQRT(1.0+TL*TL))                                      SUBP2080
      YO   = YREC + YS(LS)                                              SUBP2090
      AY   = YO                                                         SUBP2100
      GO TO   30                                                        SUBP2110
  180 CONTINUE                                                          SUBP2120
      DPLL = DP                                                         SUBP2130
  190 CONTINUE                                                          SUBP2140
      SUM  = DPUR + FLND*DPUL + FLNE*DPLR + FLND*FLNE*DPLL               SUBP2150
      RETURN                                                            SUBP2160
      END                                                               SUBP2170
```

```
C  SEGMENT  4  --  RDMODE
CRDMODE          READ MODAL DATA FROM CARDS
C
      SUBROUTINE   RDMODE  (   IA,     NA,     NIN,     NM,     NOUT,
     *              IPRNT,     IERROR     )
C
C     *********     CARD IMAGES TO BE READ                        **********
C
CCQQPPPMNB..VALUE...  QQPPPMNB..VALUE...  QQPPPMNB..VALUE...
CCQQBBBN  ..VALUE...  QQBBBN  ..VALUE...  QQBBBN  ..VALUE...
C
C          WHERE.. CC      IS A CODE =  A , AZ, AY, OR -1
C                  QQQ     IS THE MODE NUMBER
C                  PPP     IS THE PANEL NUMBER
C                  BBB     IS THE BODY NUMBER
C                  N       IS THE POWER OF X
C                  M       IS THE POWER OF Y - ONLY FOR PANELS
C                  B       IS NB
C     *********                                                   **********
C
      DIMENSION     IA( 2, 150, 3 ),  NA(3),  IAZY(4),   ICARD(7)
C
C
C
      DATA     IAZY(1)  / 2HA  /
      DATA     IAZY(2)  / 2HAZ /
      DATA     IAZY(3)  / 2HAY /
      DATA     IAZY(4)  / 2H-1 /
C
      NM       =      0
      NA(1)    =    0
      NA(2)    =    0
      NA(3)    =    0
C
C                   READ A DATA CARD
   10 READ (  NIN, 1000 )  ICARD
C
C                   CHECK FOR CODE
      I        =    0
      DO    100     J  =  1,    3
      I        =    I  + 1
      IF (ICARD(1) .EQ. IAZY(J) )   GO TO  200
  100 CONTINUE
      IF (ICARD(1) .EQ. IAZY(4) )   GO TO  500
C               BAD CODE ON CARD
C
      WRITE ( NOUT, 1010 ) ICARD
      GO TO  10
C
C               STORE CARD DATA INTO CORRECT ARRAY
C
  200 DO     300     J  =  2,    6,    2
C
C               SKIP IF NO ENTRY
      IF (ICARD(J) .EQ. 0 )    GO    TO    300
C                            CHECK TO SEE IF NUMBER OF VALUES EQUAL 150
      IF ( NA(I) .GE. 150 ) GO TO 10
      NA (I)   =    NA (I)   +   1
      N         =    NA (I)
      IA(1, N, I )   =  ICARD ( J    )
      IA(2, N, I )   =  ICARD ( J +1)
  300 CONTINUE
C                   GO - READ THE NEXT CARD
      GO  TO        10
C
C               END OF DATA ENCOUNTERED.  SORT THE INPUT DATA.
C
  500 DO     800     I =  1, 3
      N          =  NA(I)
      IF   (  N - 1 )  800,  740,  550
  550 DO     700     J =  2, N
      L          =      J
      LM1        =      L - 1
  600 IF ( IA( 1, L, I )  -  IA( 1, LM1, I ) )  640, 620, 700
C                    SET CODE OF LM1 TO LARGE VALUE TO ELIMINATE
C                    DUPLICATE CODES, USE THE LAST ONE INPUT

  620 CONTINUE
      IA( 1, LM1, I )    =  1000000000
      GO TO  700
  640 CONTINUE
      IT                  =  IA ( 1, LM1, I )
      IA( 1, LM1, I )  =  IA ( 1, L   , I )
      IA( 1, L   , I )  =  IT
      IT                  =  IA ( 2, LM1, I )
      IA(2, LM1, I )     =  IA ( 2, L   , I )
      IA(2, L   , I )     =  IT
      L         =   LM1
      LM1       =   LM1 - 1
      IF ( LM1 )  700, 700,  600
  700 CONTINUE
      NP       =  1
      DO     720  J  =  1,N
      IF ( IA( 1, J, I )  .EQ. 1000000000    )  GO TO 720
      IA ( 1,  NP,  I )  =     IA ( 1,  J, I )
      IA ( 2,  NP,  I )  =     IA ( 2,  J, I )
      NP         =      NP  +  1
  720 CONTINUE
      NA(I)    =     NP -1
      N         =     NP - 1
C              DETERMINE THE NUMBER OF MODES INPUT
  740 NM          =      MAX0( NM,  IA( 1, N, I ) / 10**6  )
C                      PRINT INPUT DATA
C
  750 WRITE ( NOUT, 1020 )  IAZY(I)
      WRITE ( NOUT, 1030 ) ((IA( J, K, I ) , J = 1, 2), K=1, N )
  800 CONTINUE
      IF (  NM  .LE.  0 )  GO TO 900
      RETURN

C          TERMINATE CASE IF THERE ARE NO MODES INPUT
  900 IERROR    =    20
      WRITE    (  NOUT , 1040 )
      RETURN
C
 1000 FORMAT  (  A2,I8, F10.0,  2(  2X,  I8,    F10.0 )    )
 1010 FORMAT ( 57H SUBROUTINE **RDMODE** INVALID DATA CARD. - - IGNORED
     *- -       /
     *           1X, A2, 4 ( I9, F12.4 ) )
 1020 FORMAT (26H1 INPUT MODAL DATA FOR -- , A2,  3H -- )
 1030 FORMAT (  4 (I10, F12.4) )
 1040 FORMAT ( 71H SUBROUTINE **RDMODE** NUMBER OF MODES IS LESS THAN OR
     * EQUAL TO ZERO.   // 22H *** CASE SKIPPED ***    )
      END
```

```
C  SEGMENT  5  --  SB, DUMULT, DZYMAT, MUZYC, OUTCOR, ROWDYZ
CSB
C   ***   CALCULATES THE NORMALWASH ON LIFTING SURFACE BOXES AND
C         INTERFERENCE BODY ELEMENTS DUE  TO SLENDER BODY ELEMENTS
      SUBROUTINE    SB    (    A,    NM,
     *   NAY,    NAZ,    WORK,    NWORK,    IPRNT    )
C *                                                                   *
C *       ********* ***************************************************
C *                                                                   *
C *       A          ARRAY CONTAINING THE MODAL DATA READ BY          *
C *                  SUBROUTINE ROMODE                                *
C *       NM         NUMBER OF MODES                                  *
C *       NAY        NUMBER OF MODAL COEFFICIENTS FOR Y-ORIENTED      *
C *                  BODIES.                                          *
C *       NAZ        NUMBER OF MODAL COEFFICIENTS FOR Z-ORIENTED      *
C *                  BODIES.                                          *
C *       WORK       COMPLEX WORKING ARRAY                            *
C *       NWORK      LENGTH OF THE WORKING ARRAY -WORK-               *
C *                                                                   *
C *       ********* ***************************************************
C *                                                                   *
      COMPLEX  WORK
      DIMENSION      WORK (1)
      COMMON   NP            , NTP          , NB            ,
     *         NBZ           , NBY          , IZ            ,
     *         NTY           ,FMACH         , REFA          ,
     *              REFS            , CHAR          , ND
     *         NE            , NBEA  (10, 2), NSBE  (  10 ),
     *         NC    ( 100 ), NS    ( 100 ), NBARAY( 100 ),
     *         ZB    (  10 ), YB    (  10 ), XLE   (  10 ),
     *         XTE   (  10 ), AR    (  10 ), AVR   (  10 ),
     *         AO    ( 200 ), AOP   ( 200 ), XIS1  ( 200 ),
     *         XIS2  ( 200), XIJ   ( 100 ), CS    ( 210 ),
     *         YP    ( 210 ), ZP    ( 210 ), EF    ( 200 ),
     *         SG    ( 200 ), CG    ( 200 ), X     ( 500 ),
     *         XIC   ( 500 ), KR            , XM            ,
     *         DELX  ( 500 ), XLAM ( 500 ), RIA   ( 100 ),
     *         THIA  ( 100 ), TH2A  ( 100 ), NFL   (  10 ),
     *         IFLA  (30, 2), NT12  (10, 2), NAS   ( 100 ),
     *         NASB  ( 200 ), YIP   ( 100 ), ZIP   ( 100 ),
     *         DETA  (  50 ), DZET  (  50 ), NOBODY(  50 ),

     *         NPANEL(  50 ), NTO
     *        , NTZS,      NTYS
C
      DIMENSION     A(1)
C
      NUMBR   = 0
      BETA    =    SQRT ( 1.0 -  FMACH **2)
      N        =    NTP   +    NTZ  +   NTY
      NFZB    =    1
      NLZB    =         NBZ
      NFYB    =         NB   + 1 - NBY
      NLYB    =         NB
      NDZ     =    1
      NDY     =    2
      NW1     =    3
      NOUT        =    6
      NW2         =    9
      NW3         =    10
      NW4         =    11
      NW5         =    4
      REWIND   NW1
      REWIND   NW2
      REWIND   NW3
      REWIND   NW4
      REWIND   NW5
C                    CALCULATE DZ AND DY MATRICES
C
      IF ( NTZS .LE.  0  )  GO  TO   100
      REWIND  NDZ
C                          CALCULATE DZ
      IDZDY    =    0
      CALL  DZYMAT ( WORK ,    NFZB   ,   NLZB   ,   NTZS   ,   IDZDY ,
     *          NDZ       , X          , BETA
     *    , ,      IPRNT       )
C
      REWIND  NDZ
C                       CALCULATE DY
C
  100 IF ( NTYS .LE. 0 )     GO  TO   200
      REWIND  NDY
      IDZDY   =    1
      CALL   DZYMAT ( WORK ,    NFYB   ,   NLYB   ,    NTYS   ,    IDZDY ,
     *             NDY      , X        , BETA
     *         ,      IPRNT       )
C
      REWIND  NDY
C
  200 CONTINUE
C
C                CALCULATE MAX. CORE NEEDED AND DETERMINE METHOD OF
C                      CALCULATING  -DH-
      NTZY      =       NTZS + NTYS
      MAXCOR    =        N * ( 1 + NTZY )   +   2 * NTZY
      IF (  MAXCOR  .GT. NWORK   )         GO  TO  1000
      WRITE (  6,   8000 )  N,   NM, NTZS, NTYS, MAXCOR, NWORK
      WRITE (  6,   8020 )
C               ALLOCATE STORAGE LOCATIONS FOR MATRICES
C
      ISTRAT    =      1
      NFW       =      1
      NFDZ       =      NFW  +  N
      NFDY      =      NFDZ  +  N * NTZS
      NFUZ      =      NFDY  +  N * NTYS
      NFUY      =      NFUZ  +  NTZS
      NFCPZ     =      NFUY  +  NTYS
      NFCPY     =     NFCPZ+  NTZS
      NFE       =      NFCPY+  NTYS
C
C                      READ DZ AND DY FROM TAPE
C
      IF ( NTZS  .LE. 0   )  GO  TO  300
      N1        =      NFDZ
      DO     220      I  =  1, N
      CALL  READD ( WORK(N1), NTZS , NDZ )
      N1        =      N1 +  NTZS
  220 CONTINUE
  300 IF ( NTYS  .LE. 0   )  GO  TO  350
      N1        =      NFDY
      DO     320      I  =  1, N
      CALL  READD ( WORK(N1), NTYS , NDY )
      N1        =      N1 +  NTYS
  320 CONTINUE
C                    NOW CALCULATE UZ AND UY A COLUMN AT A TIME AND
C                    MULTIPLY
  350 IF ( NTZS )      360, 360,  370
  360 ASSIGN  490  TO   IZ
      GO TO 380
  370 ASSIGN   480  TO   IZ
C
  380 IF ( NTYS )      390, 390,  400
  390 ASSIGN  520  TO   IY
      GO TO        450
  400 ASSIGN   500  TO   IY
C
C
  450 DO      600       J  =  1, NM
      GO TO   IZ   , ( 480,  490   )
C
C                       CALCULATE UZ COLUMN
  480 CALL   MUZYC ( J       ,    NAZ  , 2 , NTZS    ,    NFZB   ,    NLZB   ,
     *                 NSBE  ,    KR     ,    A       ,    A       ,
     *       CHAR     , AO        , AOP       , XIS1      , XIS2     ,
     *                 AR    ,     WORK(NFUZ)         ,     WORK(NFCPZ)      )
C
C                   CALCULATIONS FOR UY COLUMN
C
  490 GO TO IY, ( 500, 520)
C
  500 CALL   MUZYC ( J    ,     NAY ,     3 , NTYS    ,    NFYB   ,    NLYB   ,
     *                 NSBE  ,    KR     ,    A       ,    A       ,
     *       CHAR     , AO        , AOP       , XIS1      , XIS2     ,
     *                 AR    ,    WORK(NFUY)         ,    WORK(NFCPY)      )
C                        PERFORM MULTIPLICATION
```

```
  520 CONTINUE
C
      IEND  =  NFE - 1
      WRITE (6, 8040) J, NTZS, NTYS,(WORK(I), I = NFUZ,IEND )
C
      GO TO ( 540,  560) , ISTRAT
  540 DO 550  I  =  1,N
  550 WORK(I)  =   CMPLX (0.0,  0.0 )
      CALL DUMULT(  1        ,  N        ,   NTZS   ,   NTYS   ,   WORK( 1 ) ,
     *                WORK( NFDZ)      ,   WORK( NFUZ )      ,
     *                WORK( NFDY)      ,   WORK( NFUY )       )
C
C         WRITE DELTA-W COLUMN ON -NW1-
      CALL RWREC  ( 0       ,  NW1    , WORK    ,  N , NUMBR)
      GO TO 590
  560 CALL RWREC ( 0, NW3, WORK(NFUZ),  NTZS , NUMBR )
      CALL  RWREC  ( 0, NW4,  WORK(NFUY), NTYS , NUMBR )
  590 CONTINUE
C      WRITE  CPZ*A AND CPY*A ON - NW2-
      CALL RWREC   ( 0       ,  NW2    ,  WORK(NFCPZ ), NTZS + NTYS,NUMBR)
C
  600 CONTINUE
      GO TO (    700,    2000),   ISTRAT
  700 REWIND  NW1
      REWIND  NW2
      REWIND  NDZ
      REWIND  NDY
      GO TO     5000
C
C          ALL THE CALCULATIONS CANNOT BE DONE IN CORE
C
 1000 WRITE ( 6, 8000 )  N,    NM,  NTZS,  NTYS,    MAXCOR
     *          ,  NWORK
      WRITE ( 6, 8030 )
      ISTRAT     =    2
      NFUZ       =    1
      NFUY       =    NFUZ   +   NTZS
      NFCPZ      =     NFUY   +   NTYS
      NFCPY      =     NFCPZ   +   NTZS
      NFE        =     NFCPY   +   NTYS
C          DO- CALCULATE UZ AND UY AND WRITE THEM ON TAPE
      GO TO  350
C        NOW WE ARE READY TO CALCULATE DELTA-W
 2000 IF (  NTZS   .LE.  0 )   GO TO 3000
      NOWTAP    =   NW5
C         ************************                   ****************
C           CALCULATE THE PARTIAL DELTA-W FROM DZ * UZ
      IF ( NTYS .LE.  0 )   NOWTAP   =   NW1
      NCOR      =    N   +   N*NTZS   +   NTZS
      IF( NCOR  .GT.  NWORK )   GO TO 2500
      WRITE   (   NOUT,    8050)
      NEW        =       1
      NFDZ       =       NEW    +   N
      NFUZ       =       NFDZ   +   N*NTZS
      NFE        =       NFUZ   +   NTZS
      REWIND   NW3
      REWIND   NOWTAP
      NI         =       NFDZ
      DO    2100   I   =   1,N
      CALL   READD ( WORK(NI),   NTZS,   NDZ   )
      NI         =       NI   +   NTZS
 2100 CONTINUE
      DO    2400   J   = 1, NM
      DO    2200   I   =   1,   N
 2200 WORK(I)   =      CMPLX(  0.0,   0.0 )
      CALL   RWREC   ( 1, NW3,  WORK(NFUZ),   NTZS , NUMBR )
      CALL   DUMULT   (  1,  N,  NTZS,  0,  WORK(1),
     *        WORK(NFDZ),  WORK(NFUZ),  0, 0)
      CALL   RWREC   (   0,   NOWTAP,   WORK(1),   N   , NUMBR )
 2400 CONTINUE
      GO  TO   2600
 2500 CONTINUE
      WRITE   (   NOUT,   8060)
      CALL   OUTCOR   ( WORK(1),  NWORK,  NTZS,     N,     NM      NW3,
     *       0,     NOWTAP,     NDZ,      IPRNT     )
 2600 REWIND   NW3
      REWIND    NDZ
      REWIND     NOWTAP
 3000 IF (   NTYS    .LE.   0 )   GO TO 5000
C        ************************                    ****************
C          CALCULATE THE PARTIAL DELTA-W FROM DY * UY
      NOWTAP     =     NW1
      NOWIN      =     0
      IF (  NTZS   .GT.   0 )   NOWIN   =   NW5
      NCOR       =      N   +   N * NTYS   +   NTYS
      IF (   NCOR   .GT.   NWORK )   GO TO 3500
      WRITE   (   NOUT,   8070)
      NEW        =      1
      NFDY       =      NEW   +   N
      NFUY       =      NFDY   +   N * NTYS
      NFE        =      NFUY   +   NTYS
      REWIND    NW4
      REWIND     NOWTAP
      NI         =      NFDY
      DO    3100   I   =   1,     N
      CALL    READD   (   WORK(NI),     NTYS,     NDY   )
      NI         =      NI   +   NTYS
 3100 CONTINUE
      ASSIGN    3150   TO  IY
      IF   (   NTZS   .GT.   0 )   ASSIGN   3250   TO  IY
      DO     3400   J   =   1,   NM
      GO   TO   IY, ( 3150,  3250 )
 3150 DO    3200   I   =   1,     N
      WORK(I)   =      CMPLX(   0.0,    0.0)
 3200 CONTINUE
      GO   TO   3300
 3250 CALL   RWREC   (   1,   NOWIN,   WORK(1),   N , NUMBR )
 3300 CALL RWREC   (   1,      NW4,      WORK(NFUY),   NTYS   , NUMBR )
      CALL     DUMULT   (   1,    N,     NTYS,     0,      WORK(1),
     *         WORK(NFDY),      WORK(NFUY),       0,    0 )
      CALL   RWREC   (   0,   NOWTAP,   WORK(1),     N   , NUMBR )
 3400 CONTINUE
      GO TO 4000
 3500 CONTINUE
      WRITE   (   NOUT,   8080)
      CALL    OUTCOR (  WORK(1),   NWORK,    NTYS,      N,      NM,       NW4,
     *      NOWIN,  NOWTAP,       NDY,        IPRNT      )
 4000 CONTINUE
      REWIND     NDY
      REWIND   NW4
      REWIND   NW1
      REWIND   NW5
 5000 DO   5100   J   =   1, NM
      CALL    RWREC   (   1,   NW1, WORK(1),      N   , NUMBR )
      WRITE   (   NOUT, 9010 )    J,   (   I,   WORK(I),   I= 1, N   )
 5100 CONTINUE
      REWIND   NW1
      RETURN
 8000 FORMAT (32H0CALCULATION OF DZ AND DY.  N = , I5, 6H NM = , I5 ,
     * 8H NTZS = , I5, 8H NTYS = , I5 / 28X, 19HMAX. CORE NEEDED = , I5,
     * 15H CORE AVAIL. =  I5 )
 8010 FORMAT (9H1 COLUMN ,I5, 13H OF DELTA- W- / (2 (I6,2E13.5) )    )
 8020 FORMAT(32H0 ALL CALCULATIONS DONE IN CORE.  )
 8030 FORMAT(32H0 CALCULATIONS ONE OUT OF CORE.  )
 8040 FORMAT (  9H0  MODE =, I10/ 9H   NTZS =, I10/ 9H   NTYS =, I10/
     *     40H   U-Z, U-Y, CP-Z*DELTA-A, CP-Y*DELTA-A //(4E12.4))
 8050 FORMAT( 35H DZ * UZ CALCULATIONS DONE IN CORE  )
 8060 FORMAT (  40H DZ * UZ CALCULATIONS DONE OUT OF CORE      )
 8070 FORMAT( 35H DY * UY CALCULATIONS DONE IN CORE  )
 8080 FORMAT (  40H DY * UY CALCULATIONS DONE OUT OF CORE      )
      END
```

```
COUMULT        COMPLEX MATRIX MULTIPLICATION
C              W =  DZ * UZ   +  DY * UY
C
      SUBROUTINE  DUMULT  (   N1    ,   N2    ,   NTZS  ,   NTYS  ,
     *   W    ,   DZ    ,   UZ    ,   DY    ,   UY       )
C
C *** *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
C
C          N1         FIRST ROW OF W TO CALCULATE             I*4
C          N2         LAST ROW OF W TO CALCULATE              I*4
C          NTZS       NO. COL. IN DZ AND NO. ROWS IN UZ       I*4
C          NTYS       NO. COL. IN DY AND NO. ROWS IN UY       I*4
C          W          OUTPUT COLUMN                           COMPLEX
C          DZ         DZ MATRIX  (ROWS N1 THRU N2  )           ...
C          UZ         UZ MATRIX                                ...
C          DY         DY MATRIX  (ROWS N1 THRU N2  )           ...
C          UY         UY MATRIX                               COMPLEX
C
C *** *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
C
      COMPLEX        W(1)     , DZ (NTZS, 1 ),     UZ ( NTZS ) ,
     *                          DY (NTYS, 1 ),     UY ( NTYS )
C
      IF ( NTZS .LE. 0 ) GO   TO   100
      K       =     0
      DO      20     I  =  N1,  N2
      K       =     K  +  1
      DO      10     J  =  1,  NTZS
      W(I)    =  W(I)   +   DZ( J, K) * UZ( J )
   10 CONTINUE
   20 CONTINUE
C
C
  100 IF ( NTYS .LE. 0 )   GO  TO   200
C
C                 W  + DY * UY
      K       =     0

      DO      120     I  =  N1,
      K       =     K + 1
      DO      110     J  =  1,  NTYS
      W(I)    =  W(I)   +   DY( J, K ) * UY( J )
  110 CONTINUE
  120 CONTINUE
C
  200 RETURN
      END
```

```
CDZYMAT        CALCULATION OF DZ AND DY MATRICES
C              SLENDER BODY CALCULATIONS
      SUBROUTINE    DZYMAT  ( D      ,   NFB   ,   NLB   ,   NTZYS ,
     *   IDZDY  ,   NTAPE  ,   XP     ,    BETA
     *      , IPRNT      )
C
C *                                                                  *
C *       ********* ************************************************ *
C *                                                                  *
C *       D          WORKING ARRAY USED TO STORE A ROW OF DZ OR DY   *
C *       NFB        NUMBER OF THE FIRST BODY WITH THE ORIENTATION REQ*
C *                  UESTED                                          *
C *       NLB        NUMBER OF THE LAST BODY WITH THE ORIENTATION    *
C *                  REQUESTED                                       *
C *       NTZYS      NUMBER OF Z OR Y ORIENTED SLENDER BODY ELE.     *
C *       NTAPE      I/O UNIT NUMBER WHICH THE OUTPUT MATRIX IS TO   *
C *                  BE WRITTEN IN                                   *
C *       XP         X-CONTROL POINT COORDINATE OF LIFTING SURFACE   *
C *                  BOXES                                           *
C *       BETA       SQRT( 1.0 - M**2)                               *
C *                                                                  *
C *       ********* ************************************************ *
C *                                                                  *
C ***     *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
C ***     *   *   *   *   *   *   *   *   *   *   *   *   *   *   *
C
      COMMIN   NP            , NTP           , NB              ,
     *         NBZ           , NBY           , NTZ             ,
     *         NTY           , MACH          , REFA            ,
     *              REFS          , CBAR          , ND              ,
     *         NE            , NBEA  (10, 2), NSBE  (  10 ),         ,
     *         NC    ( 100 ), NS    ( 100 ), NBARAY( 100 ),
     *         ZB    (  10 ), YB    (  10 ), XLE   (  10 ),
     *         XTE   (  10 ), AR    (  10 ), AVR   (  10 ),
     *         AO    ( 200 ), AOP   ( 200 ), XIS1  ( 200 ),
     *         XIS2  ( 200), XIJ   ( 100 ), CS    ( 210 ),
     *         YP    ( 210 ), ZP    ( 210 ), EE    ( 200 ),
     *         SG    ( 200 ), CG    ( 200 ), K     ( 500 ),
     *         XIC   ( 500 ), KR           , XM              ,
     *         DELX  ( 500 ), XLAM ( 500 ), RIA   ( 100 ),
     *         THIA  ( 100 ), TH2A  ( 100 ), NFL   (  10 ),
     *         IFLA  (30, 2), NT12  (10, 2), NAS   ( 100 ),
     *         NASB  ( 200 ), YIP   ( 100 ), ZIP   ( 100 ),
     *         DETA  (  50 ), DZET  (  50 ), NOBODY(  50 ),
     *         NPANEL(  50 ), NTO
C
C
      DIMENSION     D(1),        XP(1)
C
      INTEGER  BY,  BZ,  C,   C1,  P,   S,   S1,  YT,  ZT,  MACH
C
      IPOT     =     6
      C1       =     0
      S1       =     0
      NFYB     =     NB - NBY   + 1
C
C                          THIS LOOP IS FOR EACH LIFTING SURF. PANEL
C
      DO      400     P  = 1,  NP
      NSP      =      NS(P)
      NCP      =      NC(P)
C                          LOOP FOR EACH STRIP IN PANEL - P -
C
      DO      350     S  = 1,  NSP
      S1       =      S1 + 1
C                    Y AND Z COORDINATE OF STRIP
      DY       =      YP(S1)
      DZ       =      ZP(S1)
      SGR      =       SG(S1)
      CGR      =       CG(S1)
C                          LOOP FOR EACH CHORDWISE ELEMENT IN STRIP
C
      DO      300     C  = 1,  NCP
      C1       =      C1  +  1
      DX       =      XP (C1)
C
C                       - ROWDYC -  CALCULATES ROW -C1- OF DZ OR DY
C
      CALL       ROWDYZ  ( NFB    ,   NLB    ,   C1     ,   NTZYS ,
     *                 D      ,   DX     ,   DY     ,   DZ     ,
     *                 BETA   ,   IDZDY  ,   NTAPE  ,   SGR    ,   CGR
     *                 IPRNT    )
C
  300 CONTINUE
C
  350 CONTINUE
C
  400 CONTINUE
C                  WE HAVE NOW CALCULATED -C1- ROWS WHICH ARE THE
C                  LIFTING SURFACES.
C            NOW....LOOP FOR THE -Z- ORIENTED BODIES
      IF ( NBZ  .LE.  0  )      GO  TO  510
      IF   ( NTZ  . LE . 0 )    GO TO   510
      SGR      =       0.0
      CGR      =       1.0
      DO      500      BZ  =  1, NBZ
      DY       =     YB (BZ )
      DZ       =     ZB (BZ )
      NBEZ       =     NBEA(BZ , 1)
C                          LOOP FOR EACH ELEMENT OF BODY - BZ -
C
      DO      450     ZT  =  1,  NBEZ
      C1       =      C1 + 1
      DX         =      XP(  C1 )
C
```

```
      CALL     ROWDYZ  ( NFB    ,   NLB    ,   C1     ,   NTZYS ,
     *                 0      ,   DX     ,   DY     ,   DZ     ,
     *                 BETA   ,   IDZDY  ,   NTAPE  ,   SGR    ,   CGR
     *                 IPRNT    )
  450 CONTINUE
  500 CONTINUE
C
C              NOW..... LOOP FOR THE - Y - ORIENTED BODIES
C

  510 IF ( NB .LT. NFYB )    GO  TO  650
      IF    ( NTY . LE . 0 )    GO TO    550
      IXP       =      NTP
      IF (   NFYB   .LE.   1 )   GO TO 530
      NFYBM1    =      NFYB   -   1
      DO     520    I =      1,   NFYBM1
  520 IXP       =      IXP   +   NBEA(   1,   I)
  530 CONTINUE
      SGR       =     -1.0
      CGR       =      0.0
      DO     600    BY   =   NFYB, NB
      DY        =      YB (BY)
      DZ        =      ZB (BY)
      NBEY      =      NBEA( BY, 1 )
C
C                                  LOOP FOR EACH ELEMENT OF BODY - BY -
      DO     550    YT     =   1,   NBEY
      C1        =      C1   +   1
      IXP       =      IXP   +   1
      DX        =      XP(   IXP   )
C
      CALL     ROWDYZ  ( NFB    ,   NLB    ,   C1     ,   NTZYS ,
     *                 0      ,   DX     ,   DY     ,   DZ     ,
     *                 BETA   ,   IDZDY  ,   NTAPE  ,   SGR    ,   CGR
     *                 IPRNT    )
C
  550 CONTINUE
  600 CONTINUE
  650 CONTINUE
C
      IF ( IPRNT   .NE.   0 )   WRITE ( NPNT, 1000)
     *     NP,     NBFZ,     NFYB,     NB,     IDZDY
      RETURN
C
 1000 FORMAT (15H DZYMAT***   NP =,I5, 11H NO. ROWS =,I5,15H FIRST Y BODY
     * = , I5,12H LAST BODY =, I5,   9H IDZDY =, I3 )
      END
```

```
CMUZYC          CALCULATION OF MU-Z AND MU-Y  COLUMNS
C               USED TO DETERMINE NORMALWASH FLOW FIELD DUE TO SLENDER
C               BODIES
C
      SUBROUTINE  MUZYC ( NMODE, NCOEF, K, NTZY, NFBODY, NLBODY, NSBE,
     *      KR     , IA     , A      , CHAR   , AO     , AOP    , XIS1   ,XIS2,
     *             XR,  UZY   ,     CPZY      )

C ***      *     *     *     *     *     *     *     *     *     *     *     *     *
C
C         NMODE        MODE NUMBER REQUESTED
C         NCOEF        NUMBER OF COEFFICIENTS INPUT BY USER
C         K            FLAG FOR TYPE OF COEFF.     = 2 - Z,   = 3 - Y
C         NTZY         NUMBER OF ELEMENTS - EITHER = NTZS OR NTYS
C         NFBODY       NUMBER OF FIRST BODY
C         NLBODY       NUMBER OF LAST BODY
C         NSBE         ARRAY OF NUMBER OF SLENDER BODY ELEMENTS PER BODY
C         KR           REDUCED FREQUENCY
C         IA           . INTEGER .  ARRAY CONTAINING IDENTIFICATION OF THE
C         A            . REAL     .  COEFFICIENT AND THE COEFF.
C         XS           SLENDER BODY MIDPOINT
C         AR           SEMI CHORD
C         WS           OUTPUT COMPLEX COLUMN
      DIMENSION      IA( 2, 150, 3), A( 2, 150, 3), IPOWR(10),NSBE(10,2),
     *               UZY( 2, 1 ),   CPZY(2, 1 ),                AO(1)     ,
     *               AOP(1) ,  XIS1(1),  XIS2(1),  AR(1)
      REAL        KR
      DATA     TWOPI / 6.283185 /
      DATA        PI / 3.141593 /
C ***  *    *     *     *     *     *     *     *     *     *     *     *     *
C            F U N C T I O N S              QQQBBPXX     Q- MODE, B- BODY
C                                                        P- POWER, X-UNUSED
      MODE(I)  =     I  / 1000000
      NBODY(I) =     MOD( I,1000000) / 1000
      NPOWR(I) =     MOD( I, 1000   ) / 100
C
C ***  *    *     *     *     *     *     *     *     *     *     *     *     *
C
C
      TWOKR          =     2.0   * KR
      TPKRC          =  TWOPI * KR / CHAR
      RAP            =  0.0
C                    SEARCH COEFFICIENT ARRAY TO FIND THE LIMITS FOR THE
C                    REQUESTED MODE
      NFA      =     1
      NLA      =   0
      DO    40    ICOEF = 1, NCOEF
      IMODE    =  MODE ( IA(1, ICOEF, K )   )
      IF (   IMODE - NMODE )   20, 30, 50
   20 NFA      =   ICOEF + 1
      GO  TO  40
C                              THE FIRST HAS BEEN FOUND, NOW FIND THE LAST
   30 NLA      =   ICOEF
   40 CONTINUE
C                             NFA AND NLA ARE THE LIMITS OF THE COEFFICIENTS
C                             FOR THIS MODE           ***
C
C              ** ZE O OUT RETURN DATA ARRAYS **
   50 DO      60    I = 1,   NTZY
      UZY ( 1, I)   =   0.0
      UZY ( 2, I)   =   0.0
      CPZY(1, I)    =      0.0
      CPZY(2, I)    =      0.0
   60 CONTINUE
                     CHECK TO SEE IF THERE ARE ANY COEFFICIENTS
                     FOR THIS MODE
      IF (   (NFA .GT. NLA )   .OR.   ( NCOEF .LE. 0 )   ) RETURN

C                    CALCULATE VALUES FOR THE
C
C                    FIRST SLENDER BODY ELEMENT
      NFSBEB   =   1
      NLSBEB   =   0
      NFU        =     0
      NL         =   NFBODY - 1
      IF ( NL .LE. 0 )   GO TO 90
C                     THE FIRST BODY IS NOT = 1, WE MUST, THEREFORE
C                     DETERMINE THE LOCATION OF THE FIRST ELEMENT IN THE
C                     FIRST BODY.
      DO     70    I = 1, NL
      NLSBEB     =     NLSBEB   +   NSBE(   I,    1 )
   70 NFSBEB          =        NFSBEB   +   NSBE ( I, 1 )
C
C                    LOOP FOR EACH BODY
   80 CONTINUE
      DO      240    IB = NFBODY, NLBODY
      IF ( NFA   .GT.   NLA   )   GO TO 260
      NFAB      =   NFA
      ICOEF     =   0
      NLSBEB          =        NLSBEB   +   NSBE (IB, 1 )
C                   CALCULATE BODY NO. FOR COEFFICIENT
   90 IBODY     -   NBODY ( IA(1, NFA, K) )
C                   SEARCH FOR COEFF. FOR BODY IB
      IF (   IBODY - IB   )   100,   120,   140
C                IBODY TOO SMALL ... LOOK AT NEXT COEFFICIENT
  100 NFA         =   NFA + 1
      NFAB        =   NFA
```

```
      IF ( NFA .LE. NLA ) GO TO 90
C
C*****            NO MORE COEFFICIENTS FOR THIS MODE                ******
      RETURN
C*****                                                              ******
C                 A COEFFICIENT HAS BEEN FOUND
  120 CONTINUE
      ICOEF    =  ICOEF + 1
      IPOWR(ICOEF) =  NPOWRI (A(1, NFA, K ) )
      NFA      =  NFA + 1
      IF( NFA .LE. NLA )  GO TO 90
      GO TO    150
                              SKIP CALCULATIONS IF NO COEFFICIENTS
   90 IF ( ICOEF  .EQ.  0 )  GO TO 230
C
C                 CALCULATE TERMS FOR EACH ELEMENT IN BODY
  150 CONTINUE
      OPAR     =     1.0  +  AR(IM)
      DEN      =     TWOPI * OPAR
      IF ( K .NE. 3 )  GO TO  170
      DEN      =      DEN * AK( IG )
      RAR            =  AR(IM)
  170 CONTINUE
      IU       =     NFU
      DO    220     IT =  NESHEB, NESHEB
      IU       =     IU  +  1
      XSC      =  0.5 * ( XIS1(IT) + XIS2(IT) ) / CHAR
      WSR            =  0.0
      WSI            =  0.0
      WSPR           =  0.0

      WSPI           =  0.0
      TWOAO    =    2.0 * AO(IT) * OPAR * ( XIS2(IT) - XIS1(IT)  )
      JA       =   NFAB  -  1
      D20      =     1.0 / ( DEN * AO(IT) **2)
      FLR            =  TWOPI * AOP(IT) * RAR
      FLI            =  IPARC * AO(IT)  * RAR
      EM       =      PI    *     AO(IT)    *     RAR   /     CHAR
      IF   (  XSC   .EQ.  0.0   )  GO TO 205
C                                  LOOP FOR EACH COEFFICIENT ( POWER )
C
      DO     200     N =  1,  ICOEF
      JA       =     JA  + 1
      I        =     IPOWR( N)
      FI       =     I
      X        =       XSC **(I - 2 ) * A( 2, JA, K )
      IF   ( I .LE. 0 )   GO  TO  180
      WSPR           =       X * FI * (FI - 1.0)      +  WSPR
  180 X        =       X * XSC
      WSR            =       X * FI    +  WSR
      X        =       X * XSC
      WSI            =       X            +  WSI
  200 CONTINUE
      GO TO 218
C     LOOP WHEN XSC = 0.0 TO CHECK FOR 0.0**0 POWER
  205 DO    215  N  =  1,    ICOEF
      JA       =     JA  +  1
      I        =     IPOWR ( N )
C           IF I .GT. 2 THERE IS NO 0 POWER
      IF ( I .GT. 2 )  GO TO 215
      IF ( I - 1 )  211, 209,  207
C           I    =  2
  207 WSPR       =     2.0 * A(2, JA, K )  +  WSPR
      GO TO 215
C          I    =      1
  209 WSR          =     A( 2,  JA,  K)  +  WSR
      GO TO 215
C           I    =     0
  211 WSI          =     A(  2,  JA,  K)  +  WSI
  215 CONTINUE
  218 CONTINUE
C
      WSPI         =   WSR * TWOKR
      WSI          =   WSI * TWOKR
      UZY ( 1,  IU) =  WSR  /  D20
      UZY ( 2,  IU) =  WSI  /  D20
      CPZY( 1, IU) =  WSR * FLR  -  WSI * FLI  +  WSPR * EM
      CPZY( 2, IU) =  WSR * FLI  +  WSI * FLR  +  WSPI * EM
C           CALCULATE  DELTA-CP  * AREA
      CPZY( 1, IU) =  CPZY( 1, IU) * TWOAO
      CPZY( 2, IU) =  CPZY( 2, IU) * TWOAO
  220 CONTINUE
C                     INCREMENT ELEMENT NUMBER
  230 NFU        =   NFU  +  NSHE ( IA, 1 )
  240 NESHEB         =     NESHEB  +  NSHE ( IM, 1 )
C
C                              CALCULATIONS COMPLETED
  260 RETURN
      END
```

```
      SUBROUTINE    OUTCOR  ( WORK  ,  NWORK ,  NTS   ,  N
     *     NM,    NUTAP,    NOWIN,    NOWOUT,   NOTAP,
C    ***   GENERATES THE  OW  MATRIX  AND WRITES IT IN TAPE
C *                                                                   *
C *      ********** *************************************************
C *                                                                   *
C *      WORK      COMPLEX WORKING ARRAY                              *
C *      NWORK     LENGTH OF WORKING ARRAY WORK                       *
C *      NTS       NUMBER OF COLUMNS IN THE MATRIX O                  *
C *      N         NUMBER OF ROWS IN THE MATRIX O                     *
C *      NM        NUMBER OF COLUMNS IN THE MATRIX U                  *
C *      NUTAP     I/O UNIT NUMBER CONTAINING THE MATRIX U IN         *
C *                COLUMN SORT.                                       *
C *      NOWIN     I/O UNIT NUMBER CONTAINING THE W' MATRIX IN        *
C *                COLUMN SORT.  IF IT IS =0, THE W' MATRIX           *
C *                IS SET TO ZERO.                                    *
C *      NOWOUT    I/O UNIT NUMBER WHICH THE DELTA-W MATRIX IS        *
C *                WRITTEN ON IN COLUMN SORT.                         *
C *      NOTAP     I/O UNIT NUMBER CONTAINING THE O MATRIX            *
C *                IN ROW SORT.                                       *
C *                                                                   *
C *      ********** *************************************************
C *                                                                   *
     *      IPRNT       )
      COMPLEX   WORK
      DIMENSION  WORK (1)
      DATA   NPOT   /6/
      NUMBR  = 0
      IF (  IPRNT  .NE.  0 )
     * WRITE (  NPOT, 8010  )  NWORK,  NTS,  N,   NM,   NUTAP,   NOWIN,
     *             NOWOUT,   NOTAP
      NOAV      =     NWORK  - NTS
      NPCOL     =     N  +  NTS
C        CALCULATE THE NUMBER OF COLUMNS OF DELTA-W TO BE CALCULATED
C          AT A TIME.  NOT TO EXCEED THE NUMBER OF MODES
      NCOLS     =     NOAV   /  NPCOL
      NCOLS     =    MINO  ( NCOLS, NM  )
      IF  (  NCOLS  .LE.  0 )  GO TO 9000
      NFW       =    1
      NFU       =     NFW  +  NCOLS  *  N
      NFO       =     NFU  +  NCOLS  *  NTS
      NFCOL     =     1
      NLCOL     =     NCOLS
      REWIND   NUTAP
C        READ THE -U-Z  OR U-Y  COLUMNS FROM THE TAPE
  100 DO 200  I  =  1, NCOLS
      NI        =   ( I-1 ) * NTS  +  NFU
      CALL   RWRFC  (  1, NUTAP,  WORK ( NI)  , NTS , NUMBR )
  200 CONTINUE
C       CHECK TO SEE IF A PARTIAL DELTA-W IS TO BE READ FROM TAPE
C         OR IF IT IS TO BE SET TO ZERO.
      IF  ( NOWIN .NE. 0 ) GO TO 400
C         IT IS TO BE SET TO ZERO
      NWOS      =     NCOLS * N - 1  +  NFW
      DO  300  I  =  NFW, NWOS
      WORK (I) =   CMPLX(  0.0,  0.0 )
```

```
  300 CONTINUE
      GO TO 600
C           PARTIAL DELTA-W IS TO BE READ FROM TAPE
  400 DO  500  I = 1, NCOLS
      NOWCOL    =   ( I-1 ) * N   +  NFW
      CALL RWREC  ( 1, NOWIN,  WORK(NOWCOL),   N , NUMBR )
  500 CONTINUE
  600 DO 950   I  =  1,  N
C             READ THE -D- ROW FROM TAPE
      CALL  READ ( WORK(NFO ),  NTS,  NOTAP )
  700 CONTINUE
C           LOOP FOR EACH OF THE COLUMNS IN CORE
  800 DO   900  J  =  1, NCOLS
      NOWCOL    =   (J-1) * N   +  NFW
      NUCOL     =   (J-1) * NTS  +  NFU
C        PERFORM THE MULTIPLICATION
      CALL   DUMULT   (  1,    1,    NTS,    0,    WORK(NOWCOL),
     *      WORK(  NFO ),  WORK( NUCOL )   ,  0,    0    )
  900 CONTINUE
  950 CONTINUE
      REWIND  NOTAP
C        WRITE THE CALCULATED DELTA-W COLUMNS ON TAPE
      DO  1000    I  =  1,  NCOLS
      NOWCOL    =   ( I-1) * N   +  NFW
      CALL   RWREC   ( C, NOWOUT,  WORK(NOWCOL), N , NUMBR )
 1000 CONTINUE
C        CHECK TO SEE IF THERE ARE MORE COLUMNS TO PROCESS
      IF  ( NLCOL  .GE.  NM )  GO TO 3000
      NFCOL     =     NFCOL  +  NCOLS
      NRCOL     =     NM  -  NLCOL
      NCOLS     =     MINO (  NRCOL,  NCOLS )
      NLCOL     =     NLCOL  +  NCOLS
      GO TO 100
 3000 CONTINUE
      WRITE (  NPOT,  8000)
 3000 RETURN
 8000 FORMAT(  60H NM IS LESS THAN OR EQUAL TO ZERO--CALCULATIONS SKIPPE
     *D.            )
 4010 FORMAT ( 71H SUBROUTINE **OUTCOR** NUMBER OF MODES IS LESS THAN OR
     * EQUAL TO ZERO.    //  30H *** CALCULATIONS SKIPPED ***      )
      END

CROWDYZ         CALCULATE A ROW OF DZ OR  DY
C               SLENDER BODY
      SUBROUTINE       ROWDYZ   (NFB    ,    NLB    ,    ROW    ,
     *                 NTZYS ,   D      ,    DX     ,    DY     ,    DZ     ,
     *      SETA,      IOZDY,       NTAPE,        SGR,       CGR,
     *       IPRNT      )
C
C *** *    *     *     *     *     *     *     *     *     *     *     *     *
C *        NFB        FIRST BODY OF THE DESIRED ORIENTATION - Z OR Y -   *
C *        NLB        LAST  BODY OF THE DESIRED ORIENTATION              *
C *        ROW        ROW OF DZ OR DY BEING CALCULATED                   *
C *        NTZYS      NO. COLUMNS TO BE CALCULATED                       *
C *        D          CALCULATED ROW                                     *
C *        DX         X - COORD. OF RECEIVING POINT                      *

C *        DY         Y - ...                                            *
C *        DZ         Z - ...                                            *
C *        BETA       SQRT ( 1 - M**2 )                                  *
C *        MACH       M - MACH NO.                                       *
C *        IOZDY      FLAG REQUIRED FOR FLLD                             *
C *** *    *     *     *     *     *     *     *     *     *     *     *     *
      COMMON    NP              , NTP            , NB             ,
     *            NBZ            , NBY            , NTZ            ,
     *            NTY            , MACH           , REFA           ,
     *                REFS              , CBAR              , N)
     *            NE             , NBEA  (10, 2), NSBE  (  10 ),
     *            NC     ( 100 ), NS     ( 100 ), NBARAY( 100 ),
     *            ZB     (  10 ), YB     (  10 ), XLE    (  10 ),
     *            XTE    (  10 ), AR     (  10 ), AVR    (  10 ),
     *            A0     ( 200 ), A0P    ( 200 ), XIS1   ( 200 ),
     *            XIS2   ( 200), XI3    ( 100 ), CS     ( 210 ),
     *            YP     ( 210 ), ZP     ( 210 ), EE     ( 200 ),
     *            SG     ( 200 ), CG     ( 200 ), X      ( 500 ),
     *            XIC    ( 500 ), KR                   , XM            ,
     *            DELX   ( 500 ), XLAM  ( 500 ), XIN    ( 100 ),
     *            THIA   ( 100 ), TH2A  ( 100 ), NFL    (  10 ),
     *            IFLA   (30, 2), NT12  (10, 2), NAS    ( 100 ),
     *            NASB   ( 200 ), YIP   ( 100 ), ZIP    ( 100 ),
     *            DETA   (  50 ), DZET  (  50 ), NBODY(  50 ),
     *            NPANEL(  50 ), NTO
     *          , NTZS,      NTYS
C
C
      DIMENSION    D(  2,  NTZYS  )
      REAL         KR,  MACH
      INTEGER      B1,  T1,  B,  T, ROW
      DATA    NPOT  /6/
      DELTA   =      ND
C
      EPSLON  =      NE
      B1      =  0
      T1      =  0

      DO      200    B =  NFB,   NLB
      B1       =  B1  +  1
C
      DAR      =  AR(B)
      NSBEB    =      NSBE ( B )
      IF ( IPRNT .NE.  0 )  WRITE ( NPOT, 1310)  B, DY,YB(B), DZ, ZB(B)
C
C

C                   LOOP FOR EACH ELEMENT IN BODY -B-
      DO      140    T = 1, NSBEB
      T1        =    T1 +  1
      D( 1, T1)      =     0.0
      D( 2, T1)      =     0.0
      XI1       =    XIS1( T1 )
      XI2       =    XIS2( T1 )
      AZRO      =      A0(T1)
      ETA        =     YB( B)
      ZETA       =     ZB( B )
C       CHECK TO SEE IF CALCULATIONS ARE TO BE MADE
      IF  (  (  DY .EQ. ETA )  .AND.  ( DZ   .EQ.  ZETA )   )  GO TO 30

C
C                          ASSIGN RETURN ADDRESS FROM DZY
      ASSIGN    20  TO    JDZDY
      GO  TO    500
   20 D (1, T1)        =  DZYR
      D (2, T1)        =  DZYI
C
C                         SKIP IF NO SYMMETRY
   30 CONTINUE
      IF (  DELTA .EQ. 0.0 )   GO TO       100
      ETA              =  -YB( B)
C      IF SENDING BODY LIES ON Z AXIS.. SKIP SYMMETRY
      IF (  ETA  .EQ.  0.0  )   GO TO 100
C       CHECK TO SEE IF CALCULATIONS ARE TO BE MADE
      IF   (  (  DY .EQ. ETA )  .AND.  ( DZ  .EQ.  ZETA )   )   GO TO 50
C                                 RETURN ADDRESS FROM DZY
      ASSIGN    40   TO    JDZDY
      GO TO     500
   40 D (1, T1)       =      D(1, T1)   +  DELTA * DZYR
      D (2, T1)       =      D(2, T1)   +  DELTA * DZYI
C
   50 CONTINUE
      IF ( EPSLON .EQ. 0.0 )   GO TO       140
C                    CALC. ONLY IF DELTA AND EPSLON  NOT EQUAL ZERO
      ETA             =  -YB( B)
      ZETA            =  -ZB(B )
C     IF SENDING BODY LIES ON Z OR Y AXIS..  SKIP GRND. AND SYMMETRY
      IF   (  ZETA  .EQ.  0.0  )            GO TO 100
```

```
C        CHECK TO SEE IF CALCULATIONS ARE TO BE MADE
      IF  (  (  DY .EQ. ETA )  .AND.  ( DZ  .EQ.  ZETA )  )  GO TO 100
C                                 ASSIGN RETURN ADDRESS FROM DZY
      ASSIGN   60   TO   JDZDY
      GO  TO   500
   60 D (1,  TI)     =  D( 1, TI ) + EPSLON* DELTA * DZYR
      D (2,  TI)     =  D( 2, TI ) + EPSLON* DELTA * DZYI
C
C                                 SKIP IF NO GROUND EFFECTS
  100 IF ( EPSLON  .EQ. 0.0 ) GO   TO   140
      ETA      =      YB(B)
      ZETA     =     -ZB(B)
C     IF SENDING BODY LIES ON THE Y AXIS..  SKIP GROUND EFFECTS
      IF (  ZETA  .EQ.  0.0  )  GO TO 140
C        CHECK TO SEE IF CALCULATIONS ARE TO BE MADE
      IF  (  (  DY .EQ. ETA )  .AND.  ( DZ  .EQ.  ZETA )  )  GO TO 140
C                                   ASSIGN RETURN ADDRESS FROM DZY
C
      ASSIGN   120   TO   JDZDY
      GO  TO   500
  120 D (1, TI)      =     D(1, TI)  +  EPSLON * DZYR
      D (2, TI)      =     D(2, TI)  +  EPSLON * DZYI
  140 CONTINUE
C                            END OF LOOP FOR ELEMENT
  180 CONTINUE
C                 200 IS END OF LOOP ON SLENDER BODY
C
  200 CONTINUE
C
C                WRITE ROW ON TAPE,  ROW NUMBER, NO. ELEMENTS, DATA
      WRITE ( NTAPE )  ROW, TI, D

       IF ( IPRNT .NE. 0 )  WRITE ( NPUT, 1300)  ROW, TI, D
       RETURN
C
C                     CALL SEQUENCE TO DZY
C
  500 CALL  DZY (   DX     ,   DY     ,   DZ     ,   SGR    ,   CGR    ,
     *              XI1    ,   XI2    ,   ETA    ,   ZETA   ,   DAR    ,
     *    AZRO  ,   AR       , CBAR     , BETA       ,
     *      MACH,     IDZDY,     DZYR,     DZYI,
     *    IPRNT     )
C
C
      GO TO    JDZDY , (  20,  40,  60, 120 )
C
 1300 FORMAT (1PH RWDYZ--- ROW NO., I5, 1H,, I10,10H ELEMENTS. /
     *       ( 6E12.4)   )
 1310 FORMAT (12H ROWDYZ  3 *, I10, 4E20.9 )
 1320 FORMAT  (1H , Z10   )
      END
```

```
C  SEGMENT  6  --  WANDWT
CWANDWT                                                                  WAWT0010
C           THE W MATRIX IS FORMED, THE DELTA - W MATRIX IS READ        WAWT0020
C           FROM TAPE, AND FROM THESE THE WT MATRIX IS FORMED AND       WAWT0030
C           SAVED ON TAPE.                                              WAWT0040
C                                                                        WAWT0050
      SUBROUTINE  WANDWT  (   A     ,   IA    ,  NSARRY,   NBARAY,      WAWT0060
     *            X     ,   YP    ,   Y1    ,   ZP    ,   Z1    ,       WAWT0070
     *    NR      , CBAR   , KR     , NP     , NTP     , NM     ,       WAWT0080
     *    NCOFF,   NDW,  NWT,  NOUT,   W,  DW,   NB,                    WAWT0090
     *    IPRNT    )                                                     WAWT0100
C                                                                        WAWT0110
C *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   * WAWT0120
C *        A            ARRAY OF COEFFICIENTS                          * WAWT0130
C *        IA                                                          * WAWT0140
C *        NSARRY       NUMBER OF STRIPS PER PANEL                     * WAWT0150
C *        NBARAY       ACCUMULATIVE NO. OF BOXES IN PANELS            * WAWT0160
C *        X            3/4 - CHORD POINT OF LIFTING-SURFACE BOX       * WAWT0170
C *        YP & ZP      3/4 - CHORD POINT OF LIFTING-SURFACE STRIP     * WAWT0180
C *        Y1 & Z1      PANEL COORDINATE                               * WAWT0190
C *        BR           REFERENCE SEMI-CHORD                           * WAWT0200
C *        KR           REDUCED FREQUENCY                              * WAWT0210
C *        NP           NO. PANELS                                     * WAWT0220
C *        NTP          NO. LIFTING-SURFACE BOXES                      * WAWT0230
C *        NM           NO. MODES                                      * WAWT0240
C *        NDW          UNIT NO. FOR  DELTA - W                        * WAWT0250
C *        NWT          UNIT NO. FOR  WT                               * WAWT0260
C *        NOUT         PRINT OUTPUT UNIT                              * WAWT0270
C *        W                                                           * WAWT0280
C *        DW                                                          * WAWT0290
C *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   * WAWT0300
C                                                                        WAWT0310
      DIMENSION       A(2, 150, 3 ), IA(2, 150, 3), NSARRY(1)    ,      WAWT0320
     *                NBARAY(1)     , X(1)           , YP(1)      ,      WAWT0330
     *                Y1(1)         , ZP(1)          , Z1(1)      ,      WAWT0340
     *                W(2, NTP )    , DW( 2, NR )                        WAWT0350
C                                                                        WAWT0360
      MODE(I)        =     I / 1000000                                   WAWT0370
      NPANEL(I)      =     MOD( I , 1000000 ) / 1000                     WAWT0380
      NPOWR(I)       =     MOD( I , 1000    ) / 100                      WAWT0390
      NPOWR(I)       =     MOD( I , 100     ) / 10                       WAWT0400
      NB    (I)      =     MOD( I , 10      )                            WAWT0410
C                                                                        WAWT0420
C *** *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   *   * WAWT0430
      REAL      KR                                                       WAWT0440
C                                                                        WAWT0450
C                                                                        WAWT0460
      K         =     1                                                  WAWT0470
      NTP1      =     NTP + 1                                            WAWT0480
C                 LOOP FOR EACH MODE                                     WAWT0490
C                                                                        WAWT0500
      DO     500     NMODE = 1, NM                                       WAWT0510
C          **    ZERO OUT THE   DW   ARRAY                               WAWT0520
C                                                                        WAWT0530
      DO   10    JDW=1,NR                                                WAWT0540
      DW(1,JDW)   =  0.0                                                 WAWT0550
   10 DW(2,JDW)   =  0.0                                                 WAWT0560
C                                                                        WAWT0570
C                                                                        WAWT0580
C                     SEARCH COEFFICIENT ARRAY TO FIND THE LIMITS        WAWT0590
C                     FOR THE REQUESTED MODE                             WAWT0600
C                                                                        WAWT0610
      NFA       =     1                                                  WAWT0620
      NLA       =     1                                                  WAWT0630
      DO       40     ICOEF  =  1,  NCOFF                                WAWT0640
      IMODE     =     MODE ( IA( 1, ICOEF, K ) )                         WAWT0650
      IF ( IMODE - NMODE )  20,  30,  50                                 WAWT0660
   20 NFA       =     ICOEF   +  1                                       WAWT0670
      GO   TO    40                                                      WAWT0680
C                     THE FIRST HAS BEEN FOUND, NOW FIND THE LAST        WAWT0690
   30 NLA       =     ICOEF                                              WAWT0700
   40 CONTINUE                                                           WAWT0710
C                                                                        WAWT0720
C                     NFA AND NLA ARE THE LIMITS OF THE                  WAWT0730
C                     COEFFICIENTS FOR THIS MODE                         WAWT0740
C                                                                        WAWT0750
C              **   ZERO OUT THE RETURN ARRAY    **                      WAWT0760
C                                                                        WAWT0770
   50 DO       60     I  =  1,   NTP                                     WAWT0780
      W ( 1, I )    =    0.0                                             WAWT0790
      W ( 2, I )    =    0.0                                             WAWT0800
   60 CONTINUE                                                           WAWT0810
      NLSTRP   =     0                                                   WAWT0820
      NFSTRP   =     1                                                   WAWT0830
      NFBOX    =     1                                                   WAWT0840
C                                                                        WAWT0850
C              * *  LOOP FOR EACH PANEL  * *                             WAWT0860
      DO     340   IP  =  1,    NP                                       WAWT0870
      IF ( NFA .GT.  NLA )   GO TO 350                                   WAWT0880
      NLSTRP   =      NLSTRP +  NSARRY(IP)                               WAWT0890
      ICOEF    =      0                                                  WAWT0900
      NLBOX    =      NBARAY( IP )                                       WAWT0910
      NC          =   ( NLBOX - NFBOX + 1 ) / NSARRY(IP)                 WAWT0920
      NFAP     =      NFA                                                WAWT0930
C                     PANEL NUMBER FOR THE COEFFICIENT - NFA -           WAWT0940
C                                                                        
   90 IPANEL   =      NPANEL ( IA( 1, NFA, K ) )                         WAWT0950
C                                                                        WAWT0960
C                     SEARCH FOR A COEFFICIENT FOR THIS PANEL            WAWT0970
      IF ( IPANEL  -  IP )  100,    120,    140                          WAWT0980
C                     IPANEL TOO SMALL                                   WAWT0990
  100 NFA       =     NFA  + 1                                           WAWT1000
      NFAP      =     NFA                                                WAWT1010
      IF ( NFA .LE. NLA   )   GO TO   90                                 WAWT1020
C                                                                        WAWT1030
C****                  NO MORE COEFFICIENTS FOR THIS MODE                WAWT1040
      GO TO         350                                                  WAWT1050
C****                                                                    WAWT1060
C                      A COEFFICIENT HAS BEEN FOUND                      WAWT1070
  120 CONTINUE                                                           WAWT1080
      ICOFF    =   ICOFF + 1                                             WAWT1090
                                                                         WAWT1100
                                                                         WAWT1110
                                                                         WAWT1120
      NFA       =     NFA + 1                                            WAWT1130
      IF ( NFA   .LE. NLA )   GO TO 90                                   WAWT1140
      GO   TO    150                                                     WAWT1150
C                                                                        WAWT1160
C                      SKIP CALCULATIONS IF NO COEFFICIENTS              WAWT1170
C                                                                        WAWT1180
  140 IF ( ICOEF .EQ. 0 )   GO TO   320                                  WAWT1190
C                                                                        WAWT1200
C                      DETERMINE WHETHER TO USE RELATIVE OR ABSOLUTE COORD.WAWT1210
C                      USE THOSE SPECIFIED BY FIRST COEF. FOR PANEL      WAWT1220
  160 CONTINUE                                                           WAWT1230
      INB      =      NB ( IA( 1, NFAP,  K )  )                          WAWT1240
      IF   ( INB .EQ. 0    )     GO TO  190                              WAWT1250
      WRITE ( NOUT,   8000 )   IP                                        WAWT1260
      GO  TO    200                                                      WAWT1270
  190 WRITE ( NOUT,   8010 )   IP                                        WAWT1280
  200 CONTINUE                                                           WAWT1290
      C         =     INB                                                WAWT1300
C                     OUTER LOOP IS FOR PANEL COEF.                      WAWT1310
C                                                                        WAWT1320
      M         =     -1                                                 WAWT1330
C                                          NO. COEF. / PANEL             WAWT1340
      DO      280     IC  =  1,  ICOEF                                   WAWT1350
                                                                         WAWT1360
C                                                                        WAWT1370
      M         =     NPOWR (  IA ( 1, NFAP, K) )                        WAWT1380
      N         =     NPOWR (  IA ( 1, NFAP, K) )                        WAWT1390
      RN        =     N                                                  WAWT1400
      IF ( IPRNT .NE. 0 ) WRITE ( NOUT, 8050 )   M, N, NFAP,             WAWT1410
     *            IA(1, NFAP, K),   IA( 2, NFAP,K)                       WAWT1420
C                                                                        WAWT1430
C                     CHECK FOR CORRECT COORD. INFO.                     WAWT1440
C                                                                        WAWT1450
      IF ( NB ( IA (1, NFAP, K) ) .NE.  INB )  WRITE( NOUT, 8020) NFAP,  WAWT1460
     *            IA (1, NFAP, K),  IA (2, NFAP, K ) , INB               WAWT1470
C                                                                        WAWT1480
C                     LOOP  FOR EACH  STRIP IN PANEL - IP -              WAWT1490
C                                                                        WAWT1500
      NB1       =     NFBOX                                              WAWT1510
      NB2       =     NB1 - 1 + NC                                       WAWT1520
      DO      260     IS  =  NFSTRP,  NLSTRP                             WAWT1530
                                                                         
      Y         =     SQRT ( (YP(IS) -  C * Y1(IP) ) **2                 WAWT1540
     *                     +  (ZP(IS) -  C * Z1(IP) ) **2   )   / CBAR   WAWT1550
      YM        =     Y ** M                                             WAWT1560
  220 CONTINUE                                                           WAWT1570
      YMA       =     YM * A ( 2, NFAP, K )                              WAWT1580
      DO     240  IB  =  NB1,  NB2                                       WAWT1590
      IF (   X( IB ) .EQ. 0.0 )  GO TO 230                               WAWT1600
  225 XCS       =     X( IB ) / CBAR                                     WAWT1610
```

```
      XX          =     YMA *  XCS **( N-1)
  226 CONTINUE
      W(1,IB) =     RN * XX     +  W( 1, IB)
      W(2,IB)     =  XX * XCS   +  W(2, IB)
      GO TO  240
C       AT THIS POINT XCS IS EQUAL TO 0.0, CHECK THE VALUE OF N---
C      SET THE RESULT OF 0.0**0  EQUAL TO 1.0
  230 IF ( N-1 )  232,  234,  225
  232 W(2,IB) =       YMA    +  W(2,IB )
      GO TO   240
  234 XX          =     YMA
      XCS         =     1.0
      GO TO 226
  240 CONTINUE
      NB1         =     NB2
      NB2         =     NB1  +  NC
      NA1         =     NB1  +  1
C
  260 CONTINUE
C                       INCREMENT COEFFICIENT NUMBER
      NFAP        =    NFAP  +  1
  280 CONTINUE
C                       INCREMENT STRIP AND BOX NUMBERS
  320 NFSTRP  =         NLSTRP + 1
      NFBOX   =         NLBOX  + 1
C
  340 CONTINUE
  350 CONTINUE
      THOKR   =        2.0 * KR
      DO      360      I  =  1,  NTP
      W( 2, I)=        THOKR * W(2, I )
  360 CONTINUE
C
C          IF THERE ARE NO BODIES FOR THIS CASE, BYPASS READING OF DW
      IF   (NB.EQ.0)  GO TO  370
C                       READ  DELTA - W FROM UNIT NDW
C
      READ ( NDW )    DW
C
C
  370 CONTINUE
      DO      400      I  =  1, NTP
      DW ( 1, I)     =  W( 1, I) - DW(1, I )
      DW ( 2, I)     =  W( 2, I) - DW(2, I )
  400 CONTINUE
      DO      420      I  =  NTP1,  NR
      DW ( 1, I)     =  - DW(1, I)
      DW ( 2, I)     =  - DW(2, I)
  420 CONTINUE
C
      WRITE ( NWT )  DW

      WRITE ( NOUT, 8030 ) NMODE, W
      WRITE ( NOUT, 8040 ) NMODE, DW
C
  500 CONTINUE
      REWIND  NDW
      REWIND  NWT
      RETURN
 8000 FORMAT (37H RELATIVE COORDINATES USED FOR PANEL , I5, 1H. )
 8010 FORMAT (37H ABSOLUTE COORDINATES USED FOR PANEL , I5, 1H. )
 8020 FORMAT (65H WARNING ** INCONSISTENT COORDINANT DATA INPUT FOR COEF
     *FICIENT NO., I4, 1H-, I11, F20.8,  6H NB = ,I2, 6H USED. )
 8030 FORMAT  (12HOW FOR MODE , I5 //( 6E12.4 )  )
 8040 FORMAT   (16HO--WT-- FOR MODE     , I5//( 6E12.4)  )
 8050 FORMAT ( 30H M, N, NFAP, IA(1&2, NFAP, 1 )
     *   / 4I12, F20.5  )
      END
```

```
C  SEGMENT  7  --  SOLVIT
      SUBROUTINE SOLVIT(A    ,RA,ND,MD,KO,NI,NM,NO,NW,NPX1)
C
C   ***   SOLVES SYSTEMS OF SIMULTANEOUS LINEAR EQUATIONS  OF
C          ND   UNKNOWNS AND  MD  RIGHT HAND SIDES
C
C           ****    ***/  *******  *****  *       ***/          *
C           *   *  *  /*     *     *      *      *  /*       ******
C     ****  ****   * / *     *     ***    *      * / *  ****    *  *
C           *      */  *     *     *      *      */  *          ***
C           *      /***      *     *      *****  /***           *  *
C
C                          - C O M P L E X -
C             D I R E C T    M A T R I X    S O L U T I O N
C
C     WRITTEN BY J. L. HESS * PROGRAMMED BY T. M. RIDDELL
C
      COMPLEX     A (10000 )
C
      DIMENSION  RA ( 2,10000 )
C
C
      LOGICAL  LAST
C
      CALL TSETV
    1 FORMAT (41HITHE I4, 2H X I3, 15H MATRIX EXCEEDS I6, 7H WORDS.)
    2 FORMAT (1H0,6F20.8)
    3 FORMAT (1H0, 10HCOLUMN NO.,I6,2X,17HOF GAMMAS FOLLOWS.//)
      N = ND
      M = MD
      NM= NM
      KORE = KO / 2
      NPM = N + M
      IF (MAX0(3 * NPM, M * N) .LE. KORE) GO TO 5
      WRITE (6,  1 ) NQUAD, NCFLG, KORE
      STOP
    5 MT = NM
      REWIND MT
      NIN = NI

      REWIND NIN
      NOUT = NO
      REWIND NOUT
      MP1 = M + 1
      NN = N
      NEL = NPM
C
C - - CALCULATE THE MAXIMUM NO. OF ROWS, 'K'
C
   10 K = (KORE - NEL) / NEL
C
C - - TEST TO SEE IF THE REST OF THE MATRIX WILL FIT IN CORE
C
      LAST = K .GE. NN
      IF (LAST) K = NN
C
C - - READ 'K' ROWS OF THE AUGMENTED 'A' MATRIX
C
   30 NT = 0
      DO 40 IB = 1, K
      NS = NT + 1
      NT = NT + NEL
   40 READ (NIN) (A(ID), ID = NS, NT)
C
C - - CHECK TO SEE IF WE WERE UNLUCKY ENOUGH TO END UP WITH ONLY ONE ROW
C
      IF (K .EQ. 1) GO TO 90
C
C - - 'K' IS GREATER THAN '1' SO WE CAN START THE TRIANGULARIZATION
C
      NELP1 = NEL + 1
      NS = - NEL
      NELP2 = NELP1 + 1
C
C - - FORM THE 'TRAPEZOIDAL' ARRAY  (S)
C
      DO 55 IB = 2, K
      NP = NELP2 - IB
      NS = NS + NELP1
      NT = NS
      DO 55 ID = IB, K
      NT = NT + NEL
      MN = NT
      NB = NS
C
C     A(NT) = -A(NT) / A(NS)
C
      TEMPR = REAL(A(NT))
      DENOM = - ( REAL(A(NS)) ** 2 + AIMAG(A(NS)) ** 2 )
      RA(1,NT) = (TEMPR       * REAL(A(NS)) + AIMAG(A(NT)) *
     *              AIMAG(A(NS))) / DENOM
      RA(2,NT) = (AIMAG(A(NT)) * REAL(A(NS)) - TEMPR *
     *              AIMAG(A(NS))) / DENOM
C
      DO 50 NF = 2, NP
      MN = MN + 1
      NB = NB + 1
C
C  50 A(MN) = A(MN) + A(NT) * A(NB)
C
      RA(1,MN) = RA(1,MN) + REAL(A(NT)) * REAL(A(NB)) -
     *              AIMAG(A(NT)) * AIMAG(A(NB))
C
   50 RA(2,MN) = RA(2,MN) + REAL(A(NT)) * AIMAG(A(NB)) +
     *              AIMAG(A(NT)) * REAL(A(NB))
C
      IF (LAST) GO TO 90
C
C - - WRITE THE 'TRAPEZOIDAL' MATRIX ON TAPE
C
      NT = 0
      NP = NEL
      NS = - NEL
      DO 60 ID = 1, K
      NS = NS + NELP1
      NT = NT + NEL
      WRITE (MT) NP, (A(IB), IB = NS, NT)
   60 NP = NP - 1
      NP = NP - M
      NS = KORE - NEL + 1
C
C - - READ ANOTHER ROW
C
      DO 80 ID = 1, NP
      READ (NIN) (A(IB), IB = NS, KORE)
C
C - - MODIFY THIS ROW BY THE 'TRAPEZOIDAL' ARRAY
C
      NT = 1
      MN = NS
      DO 70 IB = 1, K
      NB = NT
      NF = MN + 1
C
C     A(MN) = -A(MN) / A(NT)
C
      TEMPR = REAL(A(MN))
      DENOM = - (REAL(A(NT)) ** 2 + AIMAG(A(NT)) ** 2)
      RA(1,MN) = (TEMPR       * REAL(A(NT)) + AIMAG(A(MN)) *
     *              AIMAG(A(NT))) / DENOM
      RA(2,MN) = (AIMAG(A(MN)) * REAL(A(NT)) - TEMPR *
     *              AIMAG(A(NT))) / DENOM
C
      DO 65 NN = NF, KORE
      NB = NB + 1
C
C  65 A(NN) = A(NN) + A(MN) * A(NB)
C
      RA(1,NN) = RA(1,NN) + REAL(A(MN)) * REAL(A(NB)) -
     *              AIMAG(A(MN)) * AIMAG(A(NB))
C
   65 RA(2,NN) = RA(2,NN) + REAL(A(MN)) * AIMAG(A(NB)) +
     *              AIMAG(A(MN)) * REAL(A(NB))
C
      MN = NF
```

```
   70 NT = NT + NELP1
C
C - - WRITE THE MODIFIED ROW ON TAPE


C
   80 WRITE (NOUT)      (A(NT), NT = NN, KORE)
      REWIND NOUT
      REWIND NIN
C
C - - SWITCH THE TAPES
C
      NT = NIN
      NIN = NOUT
      NOUT = NT
C
C - - RE-CALCULATE ROW LENGTH AND LOOP BACK
C
      NEL = NEL - K
      NN = NEL - M
      GO TO 10
C
C - - REWIND ALL TAPES
C
   90 REWIND  MT
      REWIND NIN
      REWIND NOUT
C
C - - CONDENSE THE MATRIX
C
      NN = NEL
      NL    = NEL + 1
      IF (K .EQ. 1) GO TO 105
      NS = 1
      NT = NEL
      DO 100 IB = 2, K
      NS = NS + NELP1
      NT = NT + NEL
      DO 100 IO = NS, NT
      A(NL) = A(IO)
  100 NL = NL + 1
  105 NI = KORE - K * M + 1
C
C - - THERE, NOW WE CAN START THE BACK-SOLUTION
C * * NOTE..THE FIRST AVAILABLE LOCATION FOR THE SOLUTIONS IS A(NI)
C
      NREM = N
      NEL = NPM
      LAST = K .EQ. N
      NPASS = 0
C
C - - SOLVE FOR THE ANSWERS CORRESPONDING TO 'K' ROWS
C
  110 KM1 = K - 1
      KP1 = K + 1
      NS = NL - MP1
      NPASS = NPASS + 1
      DO 130 MN = 1, M
      NF = NS + MN
C
C     A(NF) = A(NF) / A(NS)
C
      TEMPR = REAL(A(NF))
      DENOM = REAL(A(NS)) ** 2 + AIMAG(A(NS)) ** 2

     RA(1,NF) = (TEMPR        * REAL(A(NS)) + AIMAG(A(NF)) *
    *              AIMAG(A(NS))) / DENOM
     RA(2,NF) = (AIMAG(A(NF)) * REAL(A(NS)) - TEMPR * '
    *              AIMAG(A(NS))) / DENOM
C
      NT = NS
      IF (KM1 .EQ. 0) GO TO 130
      DO 125 IB = 1, KM1
      NF = NF - IB - M
      NT = NT - MP1 - IB
C
C     SUM = 0.0
C
      SUMR = 0.0
      SUMI = 0.0
C
      NP = NF
      N2 = MP1 + IB
      DO 12. IO = 1, IB
      NN = NI + IO
      NP = NP + N2 - IO
C
C 120 SUM = SUM + A(NN) * A(NP)
C
      SUMR = SUMR + REAL(A(NN)) * REAL(A(NP)) - AIMAG(A(NN)) *
    *          AIMAG(A(NP))
C
  120 SUMI = SUMI + REAL(A(NN)) * AIMAG(A(NP)) + AIMAG(A(NN)) *
    *          REAL(A(NP))
C
C 125 A(NF) = (A(NF) - SUM) / A(NT)
C
      DENOM = REAL(A(NT)) ** 2 + AIMAG(A(NT)) ** 2
C
      TEMPR = REAL(A(NF)) - SUMR
      TEMPI = AIMAG(A(NF)) - SUMI
C
      RA(1,NF) = (TEMPR * REAL(A(NT)) + TEMPI * AIMAG(A(NT))) / DENOM
  125 RA(2,NF) = (TEMPI * REAL(A(NT)) - TEMPR * AIMAG(A(NT))) / DENOM
C
  130 CONTINUE
C
C - - MOVE THE SOLUTIONS TO CONTIGUOUS LOCATIONS STARTING AT A(NI)
C
      NI = KORE + 1
      DO 140 NN = 1, K
      DO 135 MN = 1, M
      NL = NL - 1
      NI = NI - 1
  135 A(NI) = A(NL)
  140 NL = NL - NN
C
C - - WRITE THE SOLUTIONS ON TAPE
C
      WRITE (NIN) K
      NS = NI - 1
      DO 145 MN = 1, M
      NT = NS + MN
  145 WRITE ( NIN ) (A(IO), IO = NT, KORE, M)

C
C - - TEST IF THIS IS THE LAST PASS
C
      IF (LAST) GO TO 200
C
C - - WE MUST NOW MODIFY THE TRIANGULAR MATRIX TO REFLECT THE EFFECT OF
C     THE SOLUTIONS OBTAINED SO FAR  (EQ 21)
C * * NOTE..LOCATIONS A(1) TO A(NI-1) ARE NOW FREE TO USE
C
C - - CALCULATE THE NEXT VALUES OF 'NEL' AND 'NREM'
C
      NELOLD = NEL
      KOLD = K
      NEL = NEL - K
      NREM = NREM - K
C
C - - NOW APPLY THE INCREDIBLE FORMULA FOR THE NEW 'K'
C
      K = (-4 * M - 1) / 2 + IFIX(SQRT(0.25 + FLOAT((4 * M + 2) * M +
     1 2 * (KORE - NELOLD))))
      NROW = NREM - K + 1
      IF (K .LT. NREM) GO TO 150
      LAST = .TRUE.
      NROW = 1
      K = NREM
  150 NS = 1
      NT = NELOLD + 1
C
C - - READ IN THE ROWS TO BE MODIFIED
C
      DO 190 IB = 1, NREM
```

```
      NT = NT - 1                                                       SOLV3040
      IF (IB .LE. NROW) GO TO 160                                       SOLV3050
      NS = NS   NN                                                      SOLV3060
      NT = NT + NN                                                      SOLV3070
  160 READ ( MT ) NN, (A(IO), IO = NS, NT)                              SOLV3080
      NP = N1 - 1                                                       SOLV3090
      NF = NT - M - KM1                                                 SOLV3100
      NN = NN - KOLD                                                    SOLV3110
      DO 170 MN = 1, M                                                  SOLV3120
      N2 = NF                                                           SOLV3130
      NA = NP + MN                                                      SOLV3140
      NB = NA                                                           SOLV3150
C                                                                       SOLV3160
C     SUM = 0.0                                                         SOLV3170
C                                                                       SOLV3180
      SUMR = 0.0                                                        SOLV3190
      SUMI = 0.0                                                        SOLV3200
C                                                                       SOLV3210
      DO 165 IO = 1, KOLD                                               SOLV3220
C                                                                       SOLV3230
C     SUM = SUM + A(N2) * A(NA)                                         SOLV3240
C                                                                       SOLV3250
      SUMR = SUMR + REAL(A(N2)) * REAL(A(NA)) - AIMAG(A(N2)) *          SOLV3260
     *         AIMAG(A(NA))                                             SOLV3270
      SUMI = SUMI + REAL(A(N2)) * AIMAG(A(NA)) + AIMAG(A(N2)) *         SOLV3280
     *         REAL(A(NA))                                              SOLV3290
C                                                                       SOLV3300
      N2 = N2 + 1                                                       SOLV3310

  165 NA = NA + M                                                       SOLV3320
      N2 = N2 + MN - 1                                                  SOLV3330
C                                                                       SOLV3340
C 170 A(N2) = A(N2) - SUM                                               SOLV3350
C                                                                       SOLV3360
      RA(1,N2) = RA(1,N2) - SUMR                                        SOLV3370
  170 RA(2,N2) = RA(2,N2) - SUMI                                        SOLV3380
C                                                                       SOLV3390
C - - WRITE THE MODIFIED ROW ON TAPE OR CONDENSE THE ROW                SOLV3400
C                                                                       SOLV3410
      NL = NT - M + 1                                                   SOLV3420
      IF (IB .GE. NROW) GO TO 175                                       SOLV3430
      NF = NL - KP1                                                     SOLV3440
      WRITE (NOUT) NN, (A(IO), IO = NS, NF), (A(IO), IO = NL, NT)       SOLV3450
      GO TO 190                                                         SOLV3460
  175 NF = NL - KOLD                                                    SOLV3470
      DO 180 MN = NL, NT                                                SOLV3480
      A(NF) = A(MN)                                                     SOLV3490
  180 NF = NF + 1                                                       SOLV3500
  190 CONTINUE                                                          SOLV3510
      REWIND MT                                                         SOLV3520
      REWIND NOUT                                                       SOLV3530
C                                                                       SOLV3540
C - - SWITCH THE TAPES                                                  SOLV3550
C                                                                       SOLV3560
      IT = MT                                                           SOLV3570
      MT = NOUT                                                         SOLV3580
      NOUT = IT                                                         SOLV3590
C                                                                       SOLV3600
C - - LOOP BACK THRU THE SOLUTION                                       SOLV3610
C                                                                       SOLV3620
      NL = NF                                                           SOLV3630
      GO TO 110                                                         SOLV3640
C                                                                       SOLV3650
C - - START TO WRAP IT UP                                               SOLV3660
C                                                                       SOLV3670
  200 END FILE NIN                                                      SOLV3680
      REWIND NIN                                                        SOLV3690
      N2 = N                                                            SOLV3700
C                                                                       SOLV3710
C * * NOTE.. AT THIS POINT ALL LOCATIONS A(1) THRU A(KORE) ARE FREE     SOLV3720
C                                                                       SOLV3730
      DO 220 IB = 1, NPASS                                              SOLV3740
      READ (NIN) K                                                      SOLV3750
      N1 = N2 - K + 1                                                   SOLV3760
      NS = N1                                                           SOLV3770
      NT = N2                                                           SOLV3780
C                                                                       SOLV3790
C - - READ IN THE SOLUTIONS                                             SOLV3800
C                                                                       SOLV3810
      DO 210 IO = 1, M                                                  SOLV3820
      READ (NIN) (A(NN), NN = NS, NT)                                   SOLV3830
      NT = NT + N                                                       SOLV3840
  210 NS = NS + N                                                       SOLV3850
  220 N2 = N1 - 1                                                       SOLV3860
C                                                                       SOLV3870
C - - WRITE THE SOLUTIONS ON TAPE                                       SOLV3880
C                                                                       SOLV3890
      NT = 0                                                            SOLV3900

      DO 230 IO = 1, M                                                  SOLV3910
      NS = NT + 1                                                       SOLV3920
      NT = NT + N                                                       SOLV3930
      IF (NPP1.EQ.0) GO TO 230                                          SOLV3940
      WRITE (6,3) IO                                                    SOLV3950
      WRITE (6,2) (A(NN), NN = NS, NT)                                  SOLV3960
  230 WRITE (NW) (A(NN), NN = NS, NT)                                   SOLV3970
C                                                                       SOLV3980
      CALL TIMEV(BB)                                                    SOLV3990
      BB = BB/60.0                                                      SOLV4000
      WRITE (6, 300) N, N, M, BB                                        SOLV4010
  300 FORMAT (4HOTHE I5, 2H X I5, 12H MATRIX WITH I4, 35H RIGHT SIDES WASOLV4020
     1S SOLVED DIRECTLY IN F8.3, 9H MINUTES. )                          SOLV4030
      RETURN                                                            SOLV4040
      END                                                               SOLV4050
```

```
C  SEGMENT  8  --  BFSMAT, FZY2
      SUBROUTINE BFSMAT( ND    ,  NE    ,  NB    ,  NP    ,  NTP   ,BFSM0010
     * LENGTH, NTOTAL,  IO    ,  IPRNT ,  NAS   ,  FMACH ,  YB    ,BFSM0020
     *         ZB    ,  YS    ,  ZS    ,  X     ,  DELX  ,  EE    ,BFSM0030
     *         XIC   ,  SG    ,  CG    ,  AR    ,  RIA   ,  NBEA1 ,BFSM0040
     * NBEA2,  NASB  ,  NSARAY,  NCARAY,  BFS   ,  AVR   ,  CBAR  ,BFSM0050
     *         AO    ,  XIS1  ,  XIS2  ,  KR    ,  NSBEA ,  IBFS  )BFSM0060
C                                                                       BFSM0070
C-----------------------------------------------------------------------BFSM0080
C         ND           SYMMETRY FLAG                                    BFSM0090
C         NE           GROUND EFFECTS FLAG                              BFSM0100
C         NB           NUMBER OF BODIES                                 BFSM0110
C         NP           NUMBER OF PANELS                                 BFSM0120
C         NTP          NUMBER OF LIFTING SURFACE BOXES                  BFSM0130
C         NTOTAL       NTP + TOTAL NO. OF Y AND Z ORIENTED BODY ELEMENTS BFSM0140
C         LENGTH       NTOTAL + THE TOTAL NUMBER OF Z- AND Y-ORIENTED   BFSM0150
C                      SLENDER BODY ELEMENTS                            BFSM0160
C         IO           I/O UNIT NUMBER THAT THE BFS MATRIX IS TO BE WRITTENBFSM0170
C         IPRNT        PRINT FLAG FOR INTERMEDIATE RESULTS              BFSM0180
C         NAS          ARRAY CONTAINING THE NUMBER OF ASSOCIATED BODIES BFSM0190
C                      FOR EACH PANEL                                   BFSM0200
C         FMACH        MACH NUMBER                                      BFSM0210
C         YB           ARRAY OF -Y- COORDINATES OF THE BODIES           BFSM0220
C         ZB           ARRAY OF -Z- COORDINATES OF THE BODIES           BFSM0230
C         YS           ARRAY OF -Y- COORDINATES OF STRIPS AND BODIES    BFSM0240
C         ZS           ARRAY OF -Z- COORDINATES OF STRIPS AND BODIES    BFSM0250
C         X            ARRAY OF 3/4 CHORD LOCATIONS OF BOXES AND        BFSM0260
C                      1/2 CHORD FOR BODY ELEMENTS                      BFSM0270
C         DELX         ARRAY OF LENGTHS OF BOXES AND BODY ELEMENTS      BFSM0280
C         EE           ARRAY OF THE SEMI-WITH OF STRIPS                 BFSM0290
C         XIC          ARRAY OF 1/4 CHORD COORDINATES OF BOXES          BFSM0300
C         SG           ARRAY OF SINE OF STRIP DIHEDRAL ANGLE            BFSM0310
C         CG           ARRAY OF COSINE OF STRIP DIHEDRAL ANGLE          BFSM0320
C         AR           ARRAY OF RATIO OF MAJOR AXES OF BODIES           BFSM0330
C         RIA          ARRAY OF RADII OF BODY ELEMENTS                  BFSM0340
C         NBEA1        ARRAY OF NUMBER OF BODY ELEMENTS PER BODY        BFSM0350
C         NBEA2        ARRAY OF THE BODY ORIENTATION FLAGS PER BODY     BFSM0360
C         NASB         ARRAY OF THE BODIES ASSOCIATED WITH PANELS       BFSM0370
C         NSARAY       ARRAY OF THE NUMBER OF STRIPS PER PANEL          BFSM0380
C         NCARAY       ARRAY OF THE NUMBER OF CHORDWISE DIV. PER PANEL  BFSM0390
C         BFS          WORK ARRAY FOR TEMPORARY STORAGE OF THE BFS COLS. BFSM0400
C          AVR         ARRAY OF RADII OF BODIES                         BFSM0410
C         CBAR         REFERENCE CHORD                                  BFSM0420
C         AO           ARRAY OF SLENDER BODY ELEMENT RADII              BFSM0430
C         XIS1         ARRAY OF SLENDER BODY ELEMENT LEADING  EDGE COORD.S BFSM0440
C         XIS2         ARRAY OF SLENDER BODY ELEMENT TRAILING EDGE COORD.S BFSM0450
C         KR           REDUCED FREQUENCY                                BFSM0460
C         NSBEA        ARRAY OF THE NUMBER OF ELEMENTS PER SLENDER BODY BFSM0470
C         IBFS         OPTION FLAG  =  1                                BFSM0480
C                                                                       BFSM0490
C-----------------------------------------------------------------------BFSM0500
      REAL             KR                                               BFSM0510
      DIMENSION        YB  ( 1 ), ZB  ( 1 ), YS  ( 1 ), ZS  ( 1 ), X   ( 1 ), BFSM0520
     *                 DELX( 1 ), EE  ( 1 ), XIC ( 1 ), SG  ( 1 ), CG  ( 1 ), BFSM0530
     *                 AR  ( 1 ), RIA ( 1 ), AVR ( 1 ), XIS1( 1 ), XIS2( 1 ), BFSM0540
     *                 AO  ( 1 ), NAS ( 1 ), NASB( 1 ), NBEA1( 1 ), NBEA2(1) , BFSM0550
     *                 NSBEA(1) , NCARAY(1), NSARAY(1)                  BFSM0560
      COMPLEX          BFS(LENGTH,2) , FWZ, FWY, MWZ, MWY               BFSM0570
      LOGICAL             LAST                                          BFSM0580
C-----------------------------------------------------------------------BFSM0590
      IFLAG    = 0                                                      BFSM0600
      NUMBR    = 0                                                      BFSM0610
      NPOT     = 6                                                      BFSM0620
      BETA2    =     1.0  -  FMACH**2                                   BFSM0630
      IFL      =     2                                                  BFSM0640
      ICOL     =     0                                                  BFSM0650
      NTY      =     NTOTAL / 2                                         BFSM0660
      GO  TO  3000                                                      BFSM0670
C-----------------------------------------------------------------------BFSM0680
C-----------------------------------------------------------------------BFSM0690
   50 CONTINUE                                                          BFSM0700
C                    - Y - ORIENTED BODIES AS SENDING ELEMENTS          BFSM0710
      SGS      =     -1.0                                               BFSM0720
      CGS      =     0.0                                                BFSM0730
      NASD     =     0                                                  BFSM0740
      IZYFLG   =     3                                                  BFSM0750
      ASSIGN  4010   TO   IBODY                                         BFSM0760
      ASSIGN  1100   TO   ICOLMN                                        BFSM0770
      GO  TO  1000                                                      BFSM0780
  100 CONTINUE                                                          BFSM0790
C                    - LIFTING SURF. BOXES AS SENDING ELEMENTS          BFSM0800
C                                                                       BFSM0810
      IF ( NTP  .LE.  0  )   GO  TO      800                            BFSM0820
      ASSIGN  200   TO   ICOLMN                                         BFSM0830
      J        =     1                                                  BFSM0840
      JPI      =     J                                                  BFSM0850
      IBOX     =     0                                                  BFSM0860
      ISTRIP   =     0                                                  BFSM0870
C                               LOOP  FOR - PANEL -                     BFSM0880
      DO       700  ISP  =   1, NP                                      BFSM0890
      NS       =     NSARAY (ISP)                                       BFSM0900
      NC       =     NCARAY (ISP)                                       BFSM0910
      NASD     =     NAS    (ISP)                                       BFSM0920
C                               LOOP  FOR - STRIP -                     BFSM0930
      DO       600  IS   =   1, NS                                      BFSM0940
      ISTRIP   =     ISTRIP  +  1                                       BFSM0950
      DYS      =     YS ( ISTRIP )                                      BFSM0960
      DZS      =     ZS ( ISTRIP )                                      BFSM0970
      SGS      =     SG ( ISTRIP )                                      BFSM0980
      CGS      =     CG ( ISTRIP )                                      BFSM0990
      WIDTH    =     2.0 *  EE (ISTRIP )                                BFSM1000
C                                                                       BFSM1010
C                               LOOP  FOR - BOX   -                     BFSM1020
      DO       500  IB   =   1, NC                                      BFSM1030
      IBOX     =     IBOX  +  1                                         BFSM1040
      DXS      =     XIC ( IBOX )                                       BFSM1050
C                                                                       BFSM1060
      ICOL     =     ICOL   +  1                                        BFSM1070
      GO TO   4000                                                      BFSM1080
  200 CONTINUE                                                          BFSM1090
      AREA     =     WIDTH  *  DELX (IBOX)                              BFSM1100
      BFS(ICOL, 1)   =     BFS(ICOL, 1)  *   AREA                       BFSM1110
      BFS(ICOL, 2)   =     BFS(ICOL,  2) *   AREA                       BFSM1120
  500 CONTINUE                                                          BFSM1130
  600 CONTINUE                                                          BFSM1140
  700 CONTINUE                                                          BFSM1150
C                - Z -   ORIENTED BODIES AS SENDING ELEMENTS            BFSM1160
C                                                                       BFSM1170
  800 CONTINUE                                                          BFSM1180
      SGS      =     0.0                                                BFSM1190
      CGS      =     1.0                                                BFSM1200
      NASD     =     0                                                  BFSM1210
      IZYFLG   =     1                                                  BFSM1220
```

```
      ASSIGN   50  TO  IBODY
      ASSIGN  1100  TO   ICOLMN
      GO  TO    1000
C-----------------------------------------------------------------------
C-----------------------------------------------------------------------
C
C   ***    LOOP FOR EACH  INTERFERENCE BODY SENDING ELEMENT
C
 1000 CONTINUE
      INDEX     =     NTP
C                              -- ISB --  IS THE SENDING BODY
      DO  1900 ISB  =   1, NB
      IF  (NBEA2(ISB) . EQ . 2 )  GO TO  1070
      IF  (NBEA2(ISB) . NE . IZYFLG)  GO TO  1900
 1070 DYS       =     YB    (ISB)
      NSBE      =     NBEA1 (ISB)
      JPI       =     1
      LAST      =     .FALSE.
      DZS       =     ZB    (ISB)
C                              -- ISBE -- IS THE ELEMENT OF THE SEND. BODY
      EARG2     =  1.0
      DO  1800 ISBE =   1, NSBE
      EARG1     =   EARG2
      INDEX     =     INDEX  +  1
      DXS       =     X (INDEX) - DELX(INDEX) /4.0
      EARG2     =  KR * DELX(INDEX) / CBAR
C                    CALCULATE THIS COLUMN
      ICOL      =     ICOL  +  1
      EIKJ1     =  CMPLX ( COS(EARG1), -SIN(EARG1) )
      EIKJ2     =  CMPLX ( COS(EARG2),  SIN(EARG2) )
      GO  TO  4000
 1100 CONTINUE
C               IS THIS THE FIRST COLUMN, YES  BRANCH
      IF ( ISBE  .EQ.  1 )  GO TO 1800
      BFS(ICOL-1, 1) = BFS(ICOL-1, 1)*EIKJ1 - BFS(ICOL, 1)*EIKJ2
      BFS(ICOL-1, 2) = BFS(ICOL-1, 2)*EIKJ1 - BFS(ICOL, 2)*EIKJ2
 1800 CONTINUE
 1900 CONTINUE
C               RETURN TO CALLING POINT - EITHER Y OR Z SENDING BODY ELEM
C
C   ***    GO  EITHER TO THE  Y-ORIENTED INTERFERENCE BODY ELEMENT LOOP
C           OR  TO THE LOOP FOR SLENDER BODY SENDING ELEMENTS
C
      GO TO  IBODY,   ( 50 , 4010)
C                    CALCULATE EACH ROW OF THE SENDING COLUMN
C-----------------------------------------------------------------------
C-----------------------------------------------------------------------
 3000 CONTINUE
      IY        =     0
      IROW      =     0
C                              -- IRB --  IS THE RECEIVING BODY
      DO  3900 IRB  =   1,  NB
      NRBE      =     NSBEA(IRB)
C
      XYB       =  YB( IRB )
      XZB       =  ZB( IRB )
C
C                              -- IRBE -- IS THE ELEM. OF THE REC. BODY
      DO  3800 IRBE =   1,  NRBE
      IY        =     IY  +  1
      IROW      =     IROW  +  2
      ORIA      =     AO  ( IY )
      DXLE      =     XIS1( IY )
      DXTE      =     XIS2( IY )
      XX1       =  DXLE
      XX2       =  DXTE
      XAA       =  ORIA
      ICOL      =     0
      GO  TO  100
 3100 CONTINUE
      CALL        RWREC( IFLAG,  IO    , BFS(1,1), LENGTH  , NUMBR   )
C
      CALL        RWREC( IFLAG,  IO    , BFS(1,2), LENGTH  , NUMBR   )
C
C     WRITE  (6,  8010)  (BFS(ITEMP, 1) ,  ITEMP = 1, LENGTH)
C     WRITE  (6,  9010)  (BFS(ITEMP, 2) ,  ITEMP = 1, LENGTH)
 8010 FORMAT  (1H0, 10X, 6E18.8)
 3800 CONTINUE
 3900 CONTINUE
      RETURN
 4000 CONTINUE
      CALL          FWMW  (  NO,   NE,     SGS,   CGS ,    IRB,    ORIA   ,
     *    AR        , DXLE      , DXTE      , YS       , ZB       , DXS     ,
     *    DYS       , DZS       , NASD      , NASB     , KR       , BETA2   ,
     *    CBAR      , AVR       , FWZ       , FWY      , MWZ      , MWY     ,
     *    IFI       , IPRNT     , IBFS      )
      BFS( ICOL   , 1)    =      FWZ * ( DXTE - DXLE )
      BFS( ICOL   , 2)    =      FWY * ( DXTE - DXLE )
      GO  TO   ICOLMN, ( 1100 ,      200   )
C-----------------------------------------------------------------------
 4010 CONTINUE
C
C   ***    LOOP FOR EACH  SLENDER      BODY SENDING ELEMENT
C
      IZYFLG    =  1
      SGS       =  0.0
      CGS       =  1.0
 4050 CONTINUE
      DO  5000  LSB = 1, NB
C   ***    LSB  IS THE INDEX OF THE SLENDER SENDING BODY
      IF  (NSBEA(LSB) . EQ . 0)   GO TO   5000
      IF  (NBEA2(LSB) . EQ . 2)   GO TO  4070
      IF  (NBEA2(LSB) . NE . IZYFLG)   GO TO   5000
 4070 CONTINUE
      XETA      =  YB( LSB )
      XZETA     =  ZB( LSB )
      MSBE      =  NSBEA(LSB)
      DO  4080  LSBE = 1, MSBE
      ICOL      =   ICOL + 1
      XXIJ      =  0.75 * XIS1(LSBE) + 0.25 * XIS2(LSBE)
      CALL          FZY2(XXIJ, XX1    , XX2   , XETA  ,  XZETA,   XYB ,   XZB  ,
     *                   XAA , BETA2, CBAR , KR    ,  FZZR ,   FZZI,   FZYR  ,
     *                   FZYI, FYZR , FYZI, FYYR, FYYI, IPRNT                )
      FWZR      =  CGS  *  FZZR + SGS  *  FZYR
      FWZI      =  CGS  *  FZZI + SGS  *  FZYI
      FWZ       =  CMPLX ( FWZR , FWZI )
      FWYR      =  CGS  *  FZYR + SGS  *  FYYR
      FWYI      =  CGS  *  FZYI + SGS  *  FYYI
      FWY       =  CMPLX ( FWYR , FWYI )
      BFS(ICOL,  1)  =  FWZ  *  ( DXTE - DXLE )
      BFS(ICOL,  2)  =  FWY  *  ( DXTE - DXLE )
 4080 CONTINUE
C     WRITE  (6, 4095)  NTP, NTOTAL, IRB, IRBE, IROW, LSB, LSBE, MSBE,
C    *                  IZYFLG, ICOL
 4095 FORMAT  (1H0, 10X, 15I5)
 5000 CONTINUE
      IF  (IZYFLG . EQ .  1)  GO TO  5010
```

```
      IZYFLG  = 3
      SGS     = -1.0
      CGS     =  0.0
      GO TO  4050
 5010 CONTINUE
C
      GO TO  3100
C
      END
```

```
      SUBROUTINE FZY2 (XIJ, X1, X2,ETA,ZETA, Y8, Z8, A, BETA2,CBAR, K,
     1              FZZR, FZZI, FZYR, FZYI, FZYR, FZYI, FYYR, FYYI, MFLG )
C   ***   THIS SUBROUTINE IS AN ALTERNATIVE TO SUBROUTINE  FMZY    ---
C         IT IS USED WHENEVER THE OPTION FLAG   IBFS  =  1
C   ***
      REAL M,K,KBAR,KBAR2,I1,I2,I3,I4,I5,I6,I7,I8,I9,I10,I11,KBAR3
      DATA     M,     KBAR, KBAR2, LASTBR / 3*0.0, 0 /
      DATA     TEST1,TEST2,CTH,STH,RAIJ,RAIJ2 /0.142857, 0.5, 1.0,3*0.0/
      DATA     CAPOR,CAPOI,I1,I2,I3,I4,I5,I6,I7,I8,I9,I10,I11/ 13*0.0 /
   10 FORMAT  (1H0, 6F20.8/)
   20 FORMAT  (1H0,10X, 12HCAPOR, CAPOI/)
   30 FORMAT  (1H0//10X, 4HXI =,F12.6, 6H  X1 =,F12.6,6H  X2 =,F12.6/
     1         10X, 5HETA =,F11.6, 8H  ZETA =,F10.6,6H  Y8 =,F12.6,
     2         6H  Z8 =, F12.6/ 10X,5HA    =,F11.6, 8H  BETA2=, F10.6,
     3         8H  CBAR =, F12.6//)
   40 FORMAT  (1H0,10X, 35HM ,KBAR, XA, DX, A2, RAIJ, CTH, STH/)
   50 FORMAT  (1H0,10X, 8HFZZR =, E20  . 8H  FZZI =, E20.8/  9X,
     1                  8H  FZYR =, E20.8,  H  FZYI =, E20.8/  9X,
     2                  8H  FYYR =, F20.8, 8H  FYYI =, E20.8/)
   60 FORMAT  (1H0,10X, 36HRWIG, RAA, CAPA, FARG, QR, QI, DELTA/)
   70 FORMAT  (1H0,10X, 43HI1, I2, I3, I4, I5, I6, I7, I8, I9, I10,I11/)
   80 FORMAT  (1H0, 10X,  3HTAU/)
   90 FORMAT  (1H0, 10X,  8HRAI, RA2/)
  100 FORMAT  (1H0,10X, 20HFTHR, FTHI, FRR, FRI/)
      IF (MFLG.NE.0) WRITE (6,30) XIJ,X1,X2,ETA,ZETA,Y8,Z8,A,BETA2,CBAR
      M        = SQRT(1.0 - BETA2)
      IF  (K. LE .0.0001 . AND . M. LE .0.0001)  GO TO  110
      KBAR    = 2.0 *K *M *A / CBAR
      KBAR2   = KBAR*KBAR
      GO TO  120
  110 CONTINUE
      KBAR    = 0.0
      KBAR2   = 0.0
  120 XA      = 0.5 * (X1 + X2)
      DX      = X2 - X1
      A2      = A * A
      EPS     = 0.001 * A2
      IF  (ETA. EQ .Y8. AND . ZETA. EQ .Z8)  GO TO  130
      RAIJ2   = (ETA-Y8)**2 + (ZETA-Z8)**2
      RAIJ    = SQRT(RAIJ2)
      CTH     = (ETA- Y8) / RAIJ
      STH     = (ZETA-Z8) / RAIJ
      IF  (RAIJ2. GT .A2)  GO TO  150
      GO TO  140
  130 CONTINUE
      RAIJ    = 0.0
      RAIJ2   = 0.0
      CTH     = 1.0
      STH     = 0.0
  140 RWIG2   = A2
      GO TO 160
  150 RWIG2   = RAIJ2
  160 RAA     = SQRT((XA -XIJ)**2 + BETA2*RWIG2)
      CT2     = CTH*CTH
      ST2     = 0.0
      IF  (ABS(STH) . GT . 0.0001)  ST2=STH*STH
      RWIG    = SQRT(RWIG2)
      IF  (MFLG.EQ.0)  GO TO  170
      WRITE  (6,40)
      WRITE  (6,10)  M, KBAR, XA, DX, A2, RAIJ, CTH, STH
  170 CONTINUE
      RAA2    = RAA * RAA
      RAA3    = RAA * RAA2
      RAA4    = RAA * RAA3
      CAPA    = M - (XA- XIJ) / RAA
      DELTA   = DX / RAA
      DELTA2  = DELTA * DELTA
      EARG    = 0.0
      IF  (KBAR . LE . 0.0001)  GO TO  180
      EARG    = KBAR * (M * (XA-XIJ) - RAA) / (BETA2 * A)
      QR      = COS(EARG) / (4.0 * DX)
      QI      = SIN(EARG) / (4.0 * DX)
      GO TO  190
  180 QR      = 1.0 / (4.0 * DX)
      QI      = 0.0
  190 CONTINUE
      IF  (MFLG.EQ.0)  GO TO  200
      WRITE  (6,60)
      WRITE  (6,10)  RWIG, RAA, CAPA, EARG, QR, QI, DELTA
  200 CONTINUE
      IF  (DELTA. GT . TEST1)  GO TO  240
      I1      = DELTA / RAA2
      TRM1    = BETA2 * A * I1
      FTHR    = A * QR * TRM1
      FTHI    = A * QI * TRM1
      IF  (KBAR. LE .0.0001)  GO TO  210
      I4      = DELTA / RAA
      TRM2    = KBAR * I4
      FTHR    = FTHR - A * QI * TRM2
      FTHI    = FTHI + A * QR * TRM2
  210 CONTINUE
      IF  (RAIJ2. GT . (A2+EPS))  GO TO  220
      FRR     = FTHR
      FRI     = FTHI
      GO TO  370
  220 I6      = DELTA / RAA4
      TRM1    = -3.0 * A2 * BETA2*BETA2 * I6
      CAPOR   = RAIJ2 * QR * TRM1
      CAPOI   = RAIJ2 * QI * TRM1
      IF  (KBAR . LE . 0.0001)  GO TO  230
      I9      = DELTA / RAA3
      TRM1    = TRM1 + KBAR2 * I1
      TRM2    = -3.0 * A * BETA2 * KBAR * I9
      CAPOR   = RAIJ2 * (QR * TRM1 - QI * TRM2)
```

```
      CAPDI   = RAIJ2 * (QR * TRM2 + QI * TRM1)                         FZY21050
  230 FRR     = FTHR + CAPDR                                            FZY21060
      FRI     = FTHI + CAPDI                                            FZY21070
      GO TO  370                                                        FZY21080
  240 CONTINUE                                                          FZY21090
      IF  (DELTA . GT . TEST2)  GO TO  320                              FZY21100
      LASTRR = 0                                                        FZY21110
      TAU     = (XA - XIJ) / RAA                                        FZY21120
      TAU2    = TAU * TAU                                               FZY21130
      I1      = DELTA * (1.0 - (-1.0+5.0*TAU2)*DELTA2/8.0) / RAA2       FZY21140
  250 TRM1    = A * BETA2 * I1                                          FZY21150
      FTHR    = A * QR * TRM1                                           FZY21160
      FTHI    = A * QI * TRM1                                           FZY21170
      IF  (KBAR . LE . 0.0001)  GO TO  270                              FZY21180
      IF  (LASTRR . NE . 0)  GO TO  350                                 FZY21190
      DELTA3 = DELTA * DELTA2                                           FZY21200
      I2      = -(TAU * DELTA3) / (4.0 * RAA)                           FZY21210
      I3      = DELTA3 / 12.0                                           FZY21220
      I4      = DELTA * (1.0 + (-1.0+3.0*TAU2)*DELTA2/12.0) / RAA       FZY21230
      I5      = -(TAU * DELTA3) / 6.0                                   FZY21240
  260 TRM1    = TRM1 - (KBAR2 * CAPA * I5) / (A * BETA2)                FZY21250
      TRM2    = KBAR * (CAPA * I2 + I4 - I3*BETA2*RWIG2/(2.0*RAA3) )    FZY21260
      FTHR    = A * (QR * TRM1 - QI * TRM2)                             FZY21270
      FTHI    = A * (QR * TRM2 + QI * TRM1)                             FZY21280
  270 IF  (RAIJ2. GT . (A2+EPS))  GO TO  290                            FZY21290
      FRR     = FTHR                                                    FZY21300
      FRI     = FTHI                                                    FZY21310
      GO TO  370                                                        FZY21320
  280 CONTINUE                                                          FZY21330
      KBAR3  = KBAR*KBAR2                                               FZY21340
      IF  (LASTRR . NE . 0)  GO TO  340                                 FZY21350
      I6      = DELTA * (1.0 + 5.0*(-1.0+7.0*TAU2)*DELTA2/24.0) / RAA4  FZY21360
  290 TRM1    = -3.0 * A2 * BETA2*BETA2 * I6                            FZY21370
      CAPDR   = RAIJ2 * QR * TRM1                                       FZY21380
      CAPDI   = RAIJ2 * QI * TRM1                                       FZY21390
      IF  (KBAR . LE . 0.0001)  GO TO  310                              FZY21400
      IF  (LASTRR . NE . 0)  GO TO  360                                 FZY21410
      I7      = -5.0 * TAU * DELTA3 / (12.0 * RAA3)                     FZY21420
      I8      = DELTA3 / (12.0 * RAA2)                                  FZY21430
      I9      = DELTA * (1.0 + (-1.0+6.0*TAU2)*DELTA2/6.0) / RAA3       FZY21440
      I10     = -DELTA3 * TAU / (3.0 * RAA2)                            FZY21450
  300 TRM1    = TRM1 + KBAR2 * (I1 + 3.0 * CAPA * I10)                  FZY21460
      TRM2    = 3.0*A*BETA2 * KBAR * (-CAPA*I7 +I8*BETA2*RWIG2/(2.0*RAA3)FZY21470
     1    -I9) + KBAR3 * CAPA * I2 / (A * BETA2)                        FZY21480
      CAPDR   = RAIJ2 * (QR * TRM1 - QI * TRM2)                         FZY21490
      CAPDI   = RAIJ2 * (QR * TRM2 + QI * TRM1)                         FZY21500
  310 FRR     = FTHR + CAPDR                                            FZY21510
      FRI     = FTHI + CAPDI                                            FZY21520
      GO TO  370                                                        FZY21530
  320 CONTINUE                                                          FZY21540
      LASTRR = 1                                                        FZY21550
      RWIG    = SQRT(RWIG2)                                             FZY21560
      RA12    = (X1 - XIJ)**2 + BETA2 * RWIG2                           FZY21570
      RA22    = (X2 - XIJ)**2 + BETA2 * RWIG2                           FZY21580
      RA1     = SQRT(RA12)                                              FZY21590
      RA2     = SQRT(RA22)                                              FZY21600
      IF  (MFLG.EQ.0)  GO TO  330                                       FZY21610
      WRITE  (6,90)                                                     FZY21620
      WRITE  (6,10)  RA1, RA2                                           FZY21630
  330 CONTINUE                                                          FZY21640
      I1      = ((X2-XIJ)/RA2 - (X1-XIJ)/RA1) / (BETA2*RWIG2)           FZY21650
      GO TO  250                                                        FZY21660
  340 CONTINUE                                                          FZY21670
      RA13    = RA1 * RA12                                              FZY21680
      RA23    = RA2 * RA22                                              FZY21690
      I6      = ((X2-XIJ)/RA23-(X1-XIJ)/RA13 + 2.0*I1)/(3.0*BETA2*RWIG2) FZY21700
      GO TO  290                                                        FZY21710
  350 PART1   = 0.5 * DX * (XA - XIJ)                                   FZY21720
      I2      = -((PART1+RAA2)/RA2 + (PART1-RAA2)/RA1)/ (BETA2*RWIG2)   FZY21730
      DENOM   = X1 - XIJ + RA1                                          FZY21740
      IF  (ABS(DENOM) . LE . 0.0001)  STOP                              FZY21750
      I11     = ALOG(ABS((X2 - XIJ + RA2) / DENOM))                     FZY21760
      I3      = I11 - 2.0*(XA - XIJ)*I2 - RAA2 * I1                     FZY21770
      DEN34   = SQRT(BETA2) * RWIG                                      FZY21780
      ARG1    = (X2 - XIJ) / DEN34                                      FZY21790
      ARG2    = (X1 - XIJ) / DEN34                                      FZY21800
      I4      = (ATAN(ARG1) - ATAN(ARG2)) / DEN34                       FZY21810
      I5      = 0.5 * ALOG(RA22 / RA12) - (XA - XIJ) * I4               FZY21820
      GO TO  260                                                        FZY21830
  360 CONTINUE                                                          FZY21840
      I7      = -(1.0/RA23 - 1.0/RA13) / 3.0 - (XA - XIJ) * I6          FZY21850
      I8      = I1 - 2.0 * (XA - XIJ) * I7 - RAA2 * I6                  FZY21860
      I9      = ((X2-XIJ)/RA22-(X1-XIJ)/RA12 + I4) / (2.0*BETA2*RWIG2)  FZY21870
      I10     = -((PART1 + RAA2)/RA22 + (PART1 - RAA2)/RA12 +           FZY21880
     1             (XA - XIJ) * I4) / (2.0 * BETA2 * RWIG2)             FZY21890
      GO TO 300                                                         FZY21900
  370 CONTINUE                                                          FZY21910
      IF  (RAIJ2. LE .(A2-EPS) . OR . MFLG.EQ.0)  GO TO 380             FZY21920
      WRITE  (6,20)                                                     FZY21930
      WRITE  (6,10)  CAPDR, CAPDI                                       FZY21940
  380 CONTINUE                                                          FZY21950
      IF  (MFLG.EQ.0)  GO TO  390                                       FZY21960
      WRITE  (6,70)                                                     FZY21970
      WRITE  (6,10)  I1, I2, I3, I4, I5, I6, I7, I8, I9, I10, I11       FZY21980
      WRITE  (6,100)                                                    FZY21990
      WRITE  (6,10)  FTHR, FTHI, FRR, FRI                               FZY22000
  390 CONTINUE                                                          FZY22010
      FZZR    = CT2 * FTHR + ST2 * FRR                                  FZY22020
      FZZI    = CT2 * FTHI + ST2 * FRI                                  FZY22030
      FYYR    = ST2 * FTHR + CT2 * FRR                                  FZY22040
      FYYI    = ST2 * FTHI + CT2 * FRI                                  FZY22050
      IF  (CTH. EQ .0.0 . OR . STH. EQ . 0.0)  GO TO  400               FZY22060
      FZYR    = CTH * STH * (FRR - FTHR)                                FZY22070
      FZYI    = CTH * STH * (FRI - FTHI)                                FZY22080
      GO TO  410                                                        FZY22090
  400 FZYR    = 0.0                                                     FZY22100
      FZYI    = 0.0                                                     FZY22110
  410 CONTINUE                                                          FZY22120
      IF  (MFLG.NE.0)  WRITE (6,50)  FZZR,FZZI,FZYR,FZYI,FYYR,FYYI      FZY22130
      RETURN                                                            FZY22140
      END                                                               FZY22150
```

```
C  SEGMENT  9  --  BFM, FMZY, ORGAN, SBLOAD
CBFM            THIS ROUTINE CALCULATE THE FORCES AND MOMENTS ON THE
C               SLENDER BODY ELEMENTS

      SUBROUTINE     BFM ( IWORK   , RWORK   , WORK     , NWORK
     *          NPTAP ,  NPSTAP ,   NBFM  ,    NPOT,
     *    NM,      IPRNT,     IERROR  ,   IBFS   )
C
C *                                                                 *
C *     ********* **************************************************
C *                                                                 *
C *     IWORK     INTEGER ARRAY FOR TEMPORARY STORAGE OF RESULTS.   *
C *               LENGTH IS 2*(NTP + NTZ + NTY + NTZS + NTYS)       *
C *     RWORK     REAL ARRAY FOR TEMPORARY STORAGE.                 *
C *               THE LENGTH IS NTP + 1 + 2*(MAX0(NTP + NTZ + NTY )
C *               ,(NTZS + NTYS ))
C *     WORK      COMPLEX ARRAY FOR TEMPORARY STORAGE OF RESULTS    *
C *               THE LENGTH IN COMPLEX WORDS REQUIRED IS           *
C *               4*( (NO. STRIPS + NBZ + NBY ) + ( NTZS + NTYS )) *
C *                                                                 *
C *     NWORK     SIZE OF THE ARRAY WORK                            *
C *     NPTAP     I/O UNIT CONTAINING THE P MATRIX                  *
C *     NPSTAP    I/O UNIT NUMBER CONTAINING THE PS MATRIX          *
C *     NBFM      I/O UNIT WHICH OUTPUT BODY FORCES AND MOMENTS     *
C *               ARE TO BE WRITTEN IN ( UNFORMATTED)               *
C *     NPOT      PRINT OUTPUT UNIT                                 *
C *     NM        NUMBER OF MODES                                   *
C *                                                                 *
C *     ********* **************************************************
C *                                                                 *
      REAL     KR
      COMPLEX       WORK,     COEF
      COMMON    NP              , NTP            , NB              ,
     *          NBZ             , NBY            , NTZ             ,
     *          NTY             , MACH           , REFA            ,
     *               REFS               , CBAR            , ND
     *          NE              , NBEA  (10, 2), NSBE  (  10 ),
     *          NCARRY( 100 ), NSARRY( 100 ), NBARAY( 100 ),
     *          Z0     (  10 ), YB     (  10 ), XLE    (  10 ),
     *          XTE    (  10 ), AR     (  10 ), AVR    (  10 ),
     *          A0     ( 200 ), A0P    ( 200 ), XIS1   ( 200 ),
     *          XIS2   ( 200), XIJ    ( 100 ), CS     ( 210 ),
     *          YP     ( 210 ), ZP     ( 210 ), EE     ( 200 ),
     *          SG     ( 200 ), CG     ( 200 ), X      ( 500 ),
     *          XIC    ( 500 ), KR              , XM              ,
     *          DELX   ( 500 ), XLAM ( 500 ), RIA   ( 100 ),
     *          TH1A   ( 100 ), TH2A   ( 100 ), NFL    (  10 ),
     *          IFLA   (10, 2), NT12   (10, 2), NAS    ( 100 ),
     *          NASB   ( 200 ), Y1P    ( 100 ), Z1P    ( 100 ),
     *          DETA   (  50 ), DZET   (  50 ), NBBODY(  50 ),
     *          NPANEL(  50 ), NTO
     *           NTZS,     NTYS
C
      DIMENSION       WORK(1),      IWORK(1),
     *                CHORD(  3  ),  RWORK(  1  )
      DATA     CHORD  /  0.75,    0.50,    0.0  /
      NUMBR   = 0
      BETA2      =   1.0   -   MACH * MACH
      DUMMY      =   0.0
C
C  ***    DUMMY = 0.0  IS A * DUMMY VARIABLE * THAT IS NOT USED IN
C             SUBROUTINE  FMZY WHEN CALLED FROM  BFM
C

      NSTRIP         =       0
      DO        10   I     =     1,    NP
      NSTRIP         =     NSARRY(I) +     NSTRIP
   10 CONTINUE
      NSBEA      =       0
      DO    15   I   =  1, NB
      NSBEA      =    NSBEA   +   NSBE(I)
   15 CONTINUE
C**********************************************************************
C              DETERMINE THE SUBSCRIPTS FOR THE DATA ARRAYS
C              TO BE USED IN THE SUBROUTINE
      NTSBE          =        NTZS +       NTYS
      NTIBE          =        NTZ  +       NTY
      NTOTAL         =        NTP  +       NTSBE   +   NTIBE
C              DATA LOCATIONS
      KSTART         =        0
      KSTOP          =        KSTART  +   NTOTAL
C
      KDELA             =        0
      KP           =      KDELA   +   NTP
C
      NFM          =      NSTRIP   +   NBZ   +   NBY
      KFWZ         =      0
      KFWY         =      KFWZ   +   NFM
      KMWZ         =      KFWY   +   NFM
      KMWY         =      KMWZ   +   NFM
      KFZ          =      KMWY   +   NFM
      KFY              =       KFZ        + NTSBE
      KMZ              =       KFY        + NTSBE
      KMY              =       KMZ        + NTSBE
      NWORD            =       KMY        + NTSBE
C
C
C              CHECK  ON CORE  AVAILABILITY
      IF   (     NWORD    .LE.    NWORK    )    GO TO   20
      WRITE (   NPOT,   8070 )   NWORD,   NWORK
      IERROR     =      10
      RETURN
   20 CONTINUE
C**********************************************************************
C
C ****              CALCULATE DELTA-A FOR LIFTING SURFACE BOXES
      N1          =     KDELA
      KS          =     0
      DO          90    JP   =   1, NP
      NC          =     NCARRY(JP)
      NS          =     NSARRY( JP )
C
      DO          80    JS   =   1, NS
C                 STRIP WIDTH
      WIDTH       =     2.0  *   EE( JS )
      DO          70    JC   =   1,NC
      KS          =     KS   +   1
      N1          =     N1 + 1
      RWORK(N1)=        DELX(KS)   * WIDTH
   70 CONTINUE
   80 CONTINUE
   90 CONTINUE
C**********************************************************************

C
      IS                 =       1
      DO    5000        IRB   =    1,     NB
      NBE               =       NSBE( IRB )
C
C              DETERMINE ELEMENT N    IN RECEIVING SLENDER BODY THAT
C              FORCES IN BOXES AND ELEMENTS MAY APPLY.
C
C                           LIFTING SURFACE BOXES
C
      N2                 =       KSTART + 1
      N3                 =       KSTOP  + 1
      CALL        ORGAN    ( IWORK (   N2 ), IWORK (   N3 ), NBE            ,
     *      NTP               , X      (  1  ), DELX   (  1   ), CHORD (  1   ),
     *      XLF    ( IRB ), XTE    ( IRB ), XIS1   ( IS   ), XIS2   ( IS   ),
     *      0,     IPRNT     )
C
      NTPS        =      NTO   -   NTP
      N1                 =       NTP    +   1
      N2                 =       KSTART + N1
      N3                 =       KSTOP  + N1
C                           INTERFERENCE BODY ELEMENTS
C
      CALL        ORGAN    ( IWORK (   N2 ), IWORK (   N3 ), NBE            ,
     *      NTPS              , X      (  N1 ), DELX   (  N1  ), CHORD (  2   ),
     *      XLF    ( IRB ), XTE    ( IRB ), XIS1   (  IS ), XIS2   ( IS   ),
     *      0,     IPRNT     )
C
```

```
      N1          =     N1    +  NTPS
      N2        =    KSTART  +  N1
      N3        =    KSTOP   +  N1
C                         SLENDER BODY ELEMENTS
C
      CALL   ORGAN  ( IWORK(N2),    IWORK(N3),     NBE,
     *    NSBEA   , XIS1  (  1  ), XIS2  (  1  ),  CHORD (  2  ),
     *    XLC   ( IRB ), XTE   ( IRB ), XIS1  ( IS  ), XIS2  ( IS  ),
     *      1,    IPRNT    )
C        ************************************************************
C
      KS              =     0
      INAS       =    1
C                   FORCE AND MOMENT FACTORS FOR LIFTING SURFACE STRIPS
C
      DO     200   JP =  1,  NP
      NC              =      NCARRY(JP )
      NS              =      NSARRY(JP )
C
C
      DO     100   JS =  1,  NS
      KS              =     KS   +  1
C
      N1              =      KFWZ + KS
      N2              =      KFWY + KS
      N3              =      KMWZ + KS
      N4              =      KMWY + KS
      ANOT     = AVR(IRB)
C
      CALL          FWMWIND , NE         , SG(KS)   , CG(KS)   , IRB       ,
     * ANOT      , AR        , DUMMY     , DUMMY    , YB        , ZB        ,

     * DUMMY    , YP(KS)    , ZP(KS)    , NAS(JP) , NASH     , KR        ,
     * BETA2    , CBAR      , AVR       , WORK(N1), WORK(N2), WORK(N3),
     * WORK(N4), NBEA(IRB, 2 )       , IPRNT    , IBFS      )
C
C
      IF ( IPRNT   .NE. 0 )
     * WRITE(NPOT, 9000)  JP,   JS,  KS,   WORK(N1),
     *      WORK  (  N2 ), WORK  (  N3 ), WORK  (  N4 )
C
C
  100 CONTINUE
      INAS        =     NAS(JP)   +   INAS
  200 CONTINUE
                          ********************
C                  FORCE AND MOMENT FACTORS FOR BODIES
C
      NZYR       =     NBEA ( IRB, 2 )
      KSZ        =     KS
      KSY        =     KSZ   +   NBZ
      DO     300       JP  = 1,  NB
      NZYFLG     =     NBEA(  JP, 2)
      IF(  NZYFLG  .LE.  2 ) GO TO 220
C               THE SENDING BODY IS A---
C       --Y-- ORIENTED BODY
  210 CGB        =     0.0
      SGB        =     -1.0
      KSY        =     KSY   +   1
      KS         =     KSY
      NZYFLG     =     3
C         CHECK TO SEE IF THE VALUES NEED TO BE CALCULATED
C            THEY WILL ONLY BE CALCULATED IF THE SENDING AND RECEIVING
C          BODIES HAVE THE SAME ORIENTATION
      IF(  NZYR - 2 )  300,      230      230
C               THE SENDING BODY IS A---
C          --Z--  ORIENTED BODY
  220 CGB        =     1.0
      SGB        =     0.0
      KSZ        =     KSZ   +   1
      KS         =     KSZ
C         CHECK TO SEE IF THE VALUES NEED TO BE CALCULATED
C            THEY WILL ONLY BE CALCULATED IF THE SENDING AND RECEIVING
C          BODIES HAVE THE SAME ORIENTATION
      IF( NZYR - 2)  230,       230,       240
  230 N1         =     KS   +   KFWZ
      N2         =     KS   +   KFWY
      N3         =     KS   +   KMWZ
      N4         =     KS   +   KMWY
      ANOT     = AVR(IRB)
      CALL          FWMWIND , NE         , SGB       , CGB       , IRB       ,
     * ANOT      , AR        , DUMMY     , DUMMY     , YB        , ZB        ,
     * DUMMY     , YB(JP)    , ZB(JP)    , 0         , NAS3      , KR        ,
     * BETA2     , CBAR      , AVR       , WORK(N1), WORK(N2), WORK(N3),
     * WORK(N4), NBEA(IRB, 2 )       , IPRNT    , IBFS      )
C
      IF ( IPRNT   .NE.  0 )
     *  WRITE(NPOT,   9000)  JP,   NBEA(IRB,2),    KS,    WORK(N1),
     *       WORK(N2),     WORK(N3),     WORK(N4)
  240 GO TO (     300,      210,       300), NZYFLG
  300 CONTINUE
C                          POSITION INPUT TAPES
C                               NPTAP - FROM -QUAS-FUTSOL-
C                               NPSTAP- FROM -SB              -
      REWIND    NPTAP
      REWIND    NPSTAP
C*************************************************************************
C
      DO    4000      MODE =    1, NM
C
      DO     400     I   =  KFZ, NWORD
      WORK   ( I + 1 ) =   0.0
  400 CONTINUE
C           READ THE -- P --  ARRAY FROM TAPE
C
      CALL      RAREC  (1, NPTAP, RWORK(KP+1), NTP*NTIRF,  NUMPR )
C        DISTRIBUTE THE LIFTING SURFACE LOADS ON THE RECEIVING
C              SLENDER BODY ELEMENTS
      KS              =  0
      NDA        =     KS    +   KDELA
      LP         =     KS    +   KP
      NJS              =  0
      DO     1000       JP   =  1, NP
      NC               =  NCARRY(JP)
      NS               =  NSARRY(JP)
      DO      900       JS   =  1, NS
      NJS              =  NJS + 1
      DO      800       JC   =  1, NC
      KS               =  KS   + 1
C                INCREMENT LP BY 2 BECAUSE THE ARRAY IS COMPLEX
      LP         =      LP     +  2
      NDA        =      NDA    +  1
      N1                =       KS   + KSTART
      IFIRST            =       IWORK  (  N1 )
      N1                =       KS   + KSTOP
      ILAST             =       IWORK  (  N1 )
C               SKIP IF LOADS ON BOX DO NOT APPLY TO S.B. ELEM.
      IF (   ILAST   .EQ.   0   )          GO TO   800
      COEF          =     CMPLX ( RWORK( LP - 1 ) * RWORK (   NDA ),
     *                          RWORK( LP       ) * RWORK (   NDA ) )
      N1               =       NJS   +  KFWZ
      N2               =       NJS   +  KFWY
      N3               =       NJS   +  KMWZ
      N4               =       NJS   +  KMWY
C                DISTRIBUTE LOADS ON BODY
      CALL   SBLOAD        ( COEF             , IFIRST           , ILAST          ,
     *      KS               , XIS1   ( IS   ), XIS2   ( IS   ), DELX   (  KS ),
     *      K       (  KS ), CHORD (   1   ), XLE    ( IRB ), XTE     ( IRB ),
     *      NBE              , WORK   (  N1 ), WORK   (  N2 ), WORK   (  N3 ),
     *      WORK  (  N4 ), WORK   (KFZ+1), WORK   (KFY+1), WORK   (KMZ+1),
     *      WORK (KMY+1),   IPRNT   )
C
  800 CONTINUE
  900 CONTINUE
 1000 CONTINUE
      WRITE ( NPOT, 8020 )
      WRITE  ( NPOT,   8060)
      N1          =     KFZ   +    1
      N2          =     KFY   +    1
      N3          =     KMZ   +    1

      N4          =     KMY   +    1
      CALL   WRTEMF ( NPOT, NBE, WORK(N1), WORK(N2),WORK(N3),WORK(N4) )
C
C*************************************************************************
C                    LOOP FOR EACH INTERFERENCE BODY
C
C          DISTRIBUTE THEINTERFERENCE BODY LOADS ON THE RECEIVING
C           BODY ELEMENTS
```

```
      IZA       =     NJS
      IYA       =     IZA  +  NKZ
      LPP       =     LP
      LPY       =     LPP  +  2 * NEZ
      KSS       =     NEZ
      [illegible]
      [illegible]
      IF ( [illegible]
      NZYFLG    =     [illegible]
      IF ( NZYFL [illegible]
C     [illegible]
 1150 CONTINUE
      IZ        =     IZA  +  1
      N1        =     IZA  +  KFWZ
      N2        =     IZA  +  KFWY
      N3        =     IZA  +  KMWZ
      N4        =     IZA  +  KMWY
      LP        =     LPZ
      LPP       =     LP   +     2 * NEZ
      KS        =     KSS
C     [illegible] THE SAME
C     [illegible]
      [illegible] 1150 ,  NZYR
 1150 CONTINUE
      IYA       =     IYA  +  1
      N1        =     IYA  +  KFWZ
      N2        =     IYA  +  KFWY
      N3        =     IYA  +  KMWZ
      N4        =     IYA  +  KMWY
      LP        =     LPY
      LPY       =     LPY  +  2 * NEY
      KS        =     KSS
      NZYFLG    =     1
C     [illegible] IF [illegible] MODES ARE
C     [illegible] SKIP CALCULATIONS IF THEY ARE NOT.
      GO TO (  1550,   1300,   1200  ),  NZYR
C
 1200 CONTINUE
      DO   1500   ISBF     =  1, NSBF
      KS        =     KS   +  1
      [illegible] =   [illegible]  +  2
      LP        =     LP   +  2
      NS        =     KSTART + KS
      IFIRST    =     IWORK ( NS )
      NS        =     KSTOP  + KS
      ILAST     =     IWORK ( NS )
C        [illegible]  =  DELTA-MU-MAX
C     FIRST ELEMENT  COEF  =  2* [illegible]
C     LAST  ELEMENT  COEF  =  [illegible] - [illegible](N-1)
C     OTHER ELEMENTS COEF  = ( [illegible](K+1) - [illegible](K-1) )/2.0
      IF (  ISBF  .EQ.  1  )  GO TO 1260
C
      IF (  NSBF  .EQ.  ISBF  )  GO TO 1270
C                  NEITHER FIRST NOR LAST ELEMENT
      COEF      =     CMPLX( RWORK(LP+1) - RWORK(LP-3),
     *                       RWORK(LP+2) - RWORK(LP-2)     ) / 2.0
      GO TO 1280
C                  FIRST   ELEMENT
 1260 COEF      =     2.0 * CMPLX( RWORK(LP-1)   ,  RWORK( LP ) )
      GO TO  1280
C                  LAST    ELEMENT
 1270 CONTINUE
      COEF      =     CMPLX( RWORK(LP-1) - RWORK(LP-3),
     *                       RWORK(LP  ) - RWORK(LP-2)     )
 1280 CONTINUE
      IF ( ILAST  .EQ.  0  )  GO TO 1500
C               DISTRIBUTE LOADS IN THE SLENDER BODY
      CALL      SBLOAD ( COEF         , IFIRST        , ILAST        ,
     *    KS            , XIS1  ( IS  ), XIS2  (  IS ), DELX  (  KS ),
     *    X     (  KS ), CHORD (  2  ), XLE   ( IR3 ), XTE   ( IR8 ),
     *    NBF           , WORK  (  N1 ), WORK  (  N2 ), WORK  (  N3 ),
     *    WORK  (  N4 ), WORK  (KFZ+1), WORK  (KFY+1), WORK  (KMZ+1),
     *    WORK (KMY+1),  IPRNT  )
C
 1500 CONTINUE
 1550 GO TO (   1600,   1150,   1600  ),  NZYFLG
 1600 KSS       =     KSS  +  NBEA
 2000 CONTINUE
      WRITE  ( [illegible] )
      WRITE  ( [illegible] )
      N1        =     KFZ  +  1
      N2        =     KFY  +  1
      N3        =     KMZ  +  1
      N4        =     KMY  +  1
      [illegible]
C
C****************************************************************************
C                  [illegible] SECTION [illegible]
C
      [illegible]
C
C        [illegible]
C        [illegible] ELEMENTS
      IZA       =     NJS
      IYA       =     IZA  +  NSZ
      KSS       =     [illegible]  +  NEZ
      LSS       =     [illegible]
C
      NPSZ      =     NP
      NPSY      =     NPSZ +  2 * NEZ
      [illegible]
      NBF       =     [illegible] ( ISN )
C     IF ( [illegible] )  GO TO [illegible]
      NZYFLG    =     [illegible] ( ISN, 2 )
      IF ( NZYFLG  .EQ.  2 )  GO TO [illegible]
 2100 CONTINUE
      IZA       =     IZA  +  1
      IF (  NZYR   .LT.  2  )  GO TO  2150
      N1        =     IZA  +  KFWZ
      N2        =     IZA  +  KFWY
      N3        =     IZA  +  KMWZ
      N4        =     IZA  +  KMWY
 2150 NPS       =     NPSZ
      NPSZ      =     NPSZ  +  2 * [illegible]
      LS        =     LSS
C
      KS        =     KSS
      GO TO (  [illegible],   2300,   [illegible]   ),  NZYR
 2200 CONTINUE
      IYA       =     IYA  +  1
      IF  (  NZYR   .LT.  2 )  GO TO 2250
      N1        =     IYA  +  KFWZ
      N2        =     IYA  +  KFWY
      N3        =     IYA  +  KMWZ
      N4        =     IYA  +  KMWY
 2250 NPS       =     NPSY
      NPSY      =     NPSY  +  2 * NBEC
      LS        =     LSS
C
      KS        =     KSS
      NZYFLG    =     3
      GO TO  (  2600,    2300,     2300  ),  NZYR
C                  ********************
C
 2300 DO       2500   ISBF     =  1,  NBF
      LS              =     LS    +  1
      KS              =     KS    +  1
      NPS       =     NPS    +  2
      NS              =     KS    + KSTART
      IFIRST          =     IWORK ( NS )
      NS              =     KS    + KSTOP
      ILAST           =     IWORK ( NS )
      IF  (  ILAST  .EQ.  0  )  GO TO 2500
      COEF      =     CMPLX (  RWORK( NPS - 1),  RWORK ( [illegible] )  )
      DELTA           =     XIS2(LS) - XIS1(LS)
 2400 CONTINUE
C                  DISTRIBUTE LOADS
      CALL      SBLOAD ( COEF         , IFIRST        , ILAST        ,
     *    LS            , XIS1  ( IS  ), XIS2  (  IS ), DELTA        ,
     *    XIS1  (  LS  ), CHORD (  3  ), XLE   ( IR3 ), XTE   ( IR8 ),
     *    NBE           , WORK  (  N1 ), WORK  (  N2 ), WORK  (  N3 ),
     *    WORK  (  N4 ), WORK  (KFZ+1), WORK  (KFY+1), WORK  (KMZ+1),
     *    WORK (KMY+1),  IPRNT  )
 2500 CONTINUE
C                       ********************
```

```
      GO  TO  (     2600,     2200,     2600  ), NZYFLG
 2600 KSS        =     KSS  +  NBEC
      LSS        =     LSS  +  NBEC
 3000 CONTINUE
      WRITE ( NPOT, 8040) IRB, MODE
      WRITE  ( NPOT, 8060 )
      N1         =     KFZ  +   1
      N2         =     KFY   +  1
      N3         =     KMZ   +  1
      N4         =     KMY   +  1
      CALL WRTFMF ( NPOT, NBE , WORK(N1),  WORK(N2), WORK(N3), WORK(N4))
C
      CALL  WRTFMU ( NBFM, MODE, NBE,  WORK(N1),    WORK(N2),
     *           WORK ( N3 ),   WORK ( N4 )    )
 4000 CONTINUE

      IS         =     IS  +  NBF
 5000 CONTINUE
      REWIND  NPTAP
      REWIND  NPSTAP
      REWIND  NBF4
      RETURN
C
C
 8000 FORMAT ( 1X,  2I4, I8, 8E11.4 )
 8010 FORMAT  (  1X, I4, 4X, I8, 8E11.4 /(17X, 8E11.4 ) )
 8020 FORMAT  (  61HLOADS IN SLENDER BODY ELEMENTS DUE TO LIFTING SURFA
     *CE BOXES.  / )
 8030 FORMAT  (  92HLOADS ON SLENDER BODY ELEMENTS DUE TO LIFTING SURFA
     *CE BOXES AND INTERFERENCE BODY ELEMENTS.  / )
 8040 FORMAT ( 33HIELEMENT LOADS FOR SLENDER BODY NUMBER ,  I3,
     *      11H.  MODE NO.  , I4 )
 8050 FORMAT  ( I5, 4F17.6   )
 8060 FORMAT ( 17X, 2HFZ, 2  , 2HFY,  22X, 2HMZ,  22X, 2HMY )
 8070 FORMAT  (  27H1SUBROUTINE ***BFM*** WORD (, I8, 25H) I, GREATER THA
     *N WORK (, I8 , 2H). //    23H *** JOB TERMINATED ***    )
      END


      SUBROUTINE    FMZY  (  DYB,  DZB,  DA,  DAR,  Y,  Z,  DKR,
     *                DM,       DCBAR,
     1                  DRFZZ,DIFZZ,DRFZY,DIFZY,DRFYZ,DIFYZ,DRFYY,DIFYY,
     2                  DRMZZ,DIMZZ,DRMZY,DIMZY,DRMYZ,DIMYZ,DRMYY,DIMYY,
     3                  IF1,IF2)
C     THIS SUBROUTINE CALCULATES  THE FORCES AND MOMENTS
C      (BOTH DIVIDED BY DYNAMIC PRESSURE) ON AN ELLIPSE WHERE
C      DA=SEMI-WIDTH
C      DAR=HIGHT/WIDTH
C      DYB,DZB = COORD. OF CENTER
C      DY  ,DZ    = COORD. OF  PRESSURE  DOUBLET
C      DKR= REDUCED FREQUENCY
C      DM = MACH NO.
C      DCBAR= REF. LENGTH
C      IF1 = ORIENTATION FLAG   FOR  Z-BODY IF1=1,  Z AND Y BODY  IF1=2
C            FOR Y-BODY  IF1 = 3
C      IF2 = REGION FLAG     IF2 = 0  IF  PRESSURE DOUBLET  OUT SIDE OF
C            BODY ,          IF2 = 1  IF  PRESSURE DOUBLET  INSIDE  BODY
C
C
C      THE OUT PUT IS GIVEN IN TERMS OF 16 VARIABLES WHICH REPRESENT
C      THE REAL AND IMAGINARY PARTS OF 8 COMPLEX  FORCES AND MOMENTS
C      4 ARE FORCES   FZZ,FZY,FYZ,FYY
C      4 ARE MOMENTS  MZZ,MZY,MYZ,MYY
C      FZZ= Z-FORCE DUE TO Z-DOUBLET , MZZ= Z-MOMENT DUE TO Z-DOUBLET
C      FZY= Z-FORCE DUE TO Y-DOUBLET , MZY= Z-MOMENT DUE TO Y-DOUBLET
C      FYZ= Y-FORCE DUE TO Z-DOUBLET , MYZ= Y-MOMENT DUE TO Z-DOUBLET
C      FYY= Y-FORCE DUE TO Y-DOUBLET , MYY= Y-MOMENT DUE TO Y-DOUBLET
C      INITIALIZATION
      DZ        =    Z
      DY        =    Y
      DRFZZ = 0.0
      DIFZZ = 0.0
      DRFZY = 0.0
      DIFZY = 0.0
      DRFYZ = 0.0
      DIFYZ = 0.0
      DRFYY = 0.0
      DIFYY = 0.0
      DRMZZ=0.0
      DIMZZ=0.0
      DRMZY=0.0
      DIMZY=0.0
      DRMYZ=0.0
      DIMYZ=0.0
      DRMYY=0.0
      DIMYY=0.0
C       SET THE REGION FLAG
      RAD        =     ( DY - DYB ) **2  +  ( DZ - DZB ) **2 / DAR**2
      IF2        =     0
      IF  (  RAD  .LE.  DA**2 )  IF2   =  1
      RKAM = DKR*DM
      IF(IF2 .EQ. 0) GO TO 400
      IF(RKAM .GT.0.001) GO TO 500
      IF(IF1 .EQ. 1) GO TO 420
      IF(IF1 .EQ. 3) GO TO 430
      DRFZZ= 1.0/(1.0 + DAR)
      DRFYY= DAR/(1.0 + DAR)
      GO TO 1000
  420 DRFZZ= 1.0/(1.0 +DAR)
      GO TO 1000
  430 DRFYY= DAR/(1.0 +DAR)
      GO TO 1000
  400 IF(RKAM .GT.0.001) GO TO 500
      IF(DAR  .NE.1.0)   GO TO 500
      IF(IF1 .EQ. 1)  GO TO 440
      IF(IF1  .EQ. 3 )  GO TO 450
      RCEN = (DY-DYB)*(DY-DYB)+ (DZ-DZB)*(DZ-DZB)
      A2R4= 0.5*DA*DA/(RCEN*RCEN)
      DRFZZ =  A2R4 * ((DY-DYB)*(DY-DYB) - (DZ-DZB)*(DZ-DZB))
      DRFZY = -A2R4 * (2.0*(DY-DYB)*(DZ-DZB))
      DRFYZ = DRFZY
      DRFYY = -DRFZZ
      GO TO 1000
  440 RCEN = (DY-DYB)*(DY-DYB)+ (DZ-DZB)*(DZ-DZB)
      A2R4= 0.5*DA*DA/(RCEN*RCEN)
      DRFZZ =  A2R4 * ((DY-DYB)*(DY-DYB) - (DZ-DZB)*(DZ-DZB))
      DRFZY = -A2R4 * (2.0*(DY-DYB)*(DZ-DZB))
      GO TO 1000
  450 RCEN = (DY-DYB)*(DY-DYB)+ (DZ-DZB)*(DZ-DZB)
      A2R4= 0.5*DA*DA/(RCEN*RCEN)
      DRFYY =  A2R4 * ((DY-DYB)*(DY-DYB) - (DZ-DZB)*(DZ-DZB))
      DRFYZ = -A2R4 * (2.0*(DY-DYB)*(DZ-DZB))
      GO TO 1000
C     NUMB  =  NUMBER OF SURFACE DIVISION FOR ELLIPSE
  500 NUMB  =  30
C     SET UP  PRELIMINARY CALCULATIONS FOR INTEGRATION  DO-LOOP
      DZ  =  DZ  -DZB
      DY  =  DY  -DYB
      IF(  ABS( 1.0 - DAR**2)  .GT. 1.0E-3)  GO TO 86
      R          =     DZ**2  +  DY**2
      SINSQ     =     DZ**2
      IF ( R  .GT.  0.000001 ) SINSQ  =  SINSQ / R

      GO TO87
   36 ARG1  =  DY*DY   +  DZ*DZ +DA*DA*(DAR*DAR  -1.0)
C     ELLIPTICAL COORDINATES ARE USED TO ESTABLISH THETOL
      SINSQ  = (SQRT (ARG1*ARG1  +4.0*DA*DA*(1.0-DAR*DAR)*DZ*DZ)-ARG1)
     1           /(2.0*DA*DA*(1.0-DAR*DAR))
C         CALCULATE THE ANGLE IN THE FIRST QUADRANT
   87 DTHETO    =      1.5707963
      IF (  SINSQ  .NE.  1.0   )
     *DTHETO    =      ARSIN(  SQRT(  SINSQ )  )
C     THE PROPER QUADRENT OF THETO MUST NOW BE ESTABLISHED
      IF( DY .GT.  0.0)  GO TO  88
      IF (DZ .GT. 0.0 )  GO TO 89
      DTHETO =  DTHETO  +  3.1415927
      GO TO 91
   99 DTHETO  =  3.1415927  - DTHETO
      GO TO 91
   38 IF (DZ  .GT.  0.0)  GO TO 91
      DTHETO  = - DTHETO
C  91 WRITE(6,1)
C   1 FORMAT(1H0,14HTHETA-SUB-ZERO)
C     WRITE(6,2) DTHETO
C   2 FORMAT(1H0,F13.6)
C     SET UP DO-LOOP  TO  INTEGRATE FORCES AND MOMENTS
C
```

```
   91 DO 1001 I=1,NUM3
C     CALC. OF ELLIPTIC COORD. THETA
      DNUM=FLOAT(NUM3)
      DTHETA= DTHETO - (FLOAT(I)-1.0)*(6.2831853/DNUM)
C     WRITE(6,3)
C   3 FORMAT(1H0, 5HTHETA)
C     WRITE(6,2) DTHETA
      DCOS = COS(DTHETA)
      DSIN = SIN(DTHETA)
C     CALC. OF POINTS ON ELLIPTIC SURFACE ,DYO=Y-COORD. ,DZO=Z-COORD.
      ETA  = DA*DCOS
      ZETA = DA*DAR*DSIN
C     WRITE(6,4)
C   4 FORMAT(1H0,  25HELLIPTIC  COORDINATES Y,Z)
C     WRITE(6,5) ETA,ZETA
C   5 FORMAT(1H0, 2F20.6)
C     CALC. OF  ELEMENT WIDTH ON ELLIPTIC SURFACE  DEF=E
      DEE = ( 3.1415927/(DNUM*DAR))* SQRT(ZETA*ZETA*DAR*DAR*DAR*DAR*ETA*
     1       ETA)
C     WRITE(6,6)
C   6 FORMAT(1H0,  13HELEMENT WIDTH)
C     WRITE(6,2) DEE
C     CALC. OF  SINE AND COS OF SURFACE  SLOPE ,SIN,COS(LAMBDA) =EF2,EF1
      EF1=( ZETA*6.2831853)/(2.0*DEE*DAR*DNUM)
      EF2=(-ETA *6.2831853*DAR )/(2.0*DEE*DNUM)
C     WRITE(6,7)
C   7 FORMAT(1H0,    41HSINE AND COSINE OF SURFACE SLOPE ,EF1,EF2)
C     WRITE(6,5) EF1,EF2
C     CALC. OF  ETA    , ZETA   , THAT IS DF1,DF2
      DF1 =    DY-ETA
      DF2 =    DZ-ZETA
C     CALC. OF  B,C,R,K
      DB=-2.0*(DF1*EF1+DF2*EF2)
      DD= EF2*DF1 - EF1*DF2
      DCC= DF1*DF1+DF2*DF2

      DRR=SQRT(DCC)
      DKK      =     2.0 * DKR * DN * DRR / DCBAR
C     WRITE(6,8)
C   8 FORMAT(1H0, 17H0B,DCC,DRR,DKK,DD)
C     WRITE(6,9) DB,DCC,DRR,DKK,DD
C   9 FORMAT(1H0,5F20.6)
C     CALCULATE THE  SUBINTEGRALS OVER A SMALL SURFACE SEGMENT FOR
C     FORCE AND MOMENT  AND FOR  Z-AND Y-ORIENTED DOUBLETS
C     FORCE AND MOMENT  AND FOR  Z-AND Y-ORIENTED DOUBLETS
C     EIM1 Y-DOUBLET ,MOMENT
C     EIM2 Z-DOUBLET ,MOMENT
C     EIF1 Y-DOUBLET ,FORCE
C     EIF2 Z-DOUBLET ,FORCE
      EIM1 =(2.0*DEE*DF1)/DRR
      EIM2 =(2.0*DEE*DF2)/DRR
C     WRITE(6,10)
C  10 FORMAT(1H0,   9HEIM1,EIM2)
C     WRITE(6,5) EIM1,EIM2
C     TWO DIFFERENT FORMULAS ARE USED FOR EIF1 AND EIF2 DEPENDING ON
C     WHETHER R/C IS LESS THAN OR GREATER THAN 1
      DTEST= DRR/ DEE
      IF(DTEST .LT. 5.0) GO TO 100
      USE1=(2.0*DEE)/DCC
      EIF1= USE1*DF1
      EIF2= USE1*DF2
C     WRITE(6,11)
C  11 FORMAT(1H0,      9HEIF1,EIF2)
C     WRITE(6,5) EIF1,EIF2
      IF(IFL .EQ. 1) GO TO 116
      IF(IFL .EQ. 3) GO TO 117
      EFZZ = -EF1*EIF2
      EFZY = -EF1*EIF1
      EFYZ =  EF2*EIF2
      EFYY =  EF2*EIF1
      GO TO  110
  116 EFZZ = -EF1*EIF2
      EFZY = -EF1*EIF1
      EFYZ = 0.0
      EFYY = 0.0
      GO TO 110
  117 EFZZ = 0.0
      EFZY = 0.0
      EFYZ =  EF2*EIF2
      EFYY =  EF2*EIF1
      GO TO 110
```

```
  100 USE5 =  SQRT(DSIN*DSIN *DAR*DAR*DCOS*DCOS)
      RCURV=  DA*USE5*USE5*USE5/DAR
      DIZZ = 1.5*EF2*EF2 -0.5
      DJZZ = EF2*(EF1 -DF2/RCURV)
      DKZZ = -EF1*DF2
C
      DIZY = 1.5*EF1*EF2
      DJZY = EF1*EF1 - EF2*DF1/RCURV
      DKZY = -EF1*DF1
C
      DIYZ =  1.5*EF1*EF2
      DJYZ = -EF2*EF2 - EF1*DF2/RCURV
      DKYZ =  EF2*DF2
C
      DIYY =  1.5*EF1*EF1-0.5
      DJYY =  -EF1*(EF2 + DF1/RCURV)
      DKYY = EF2*DF1
C
      IF(DTEST .LT. 1.5) GO TO 106
C
      IF(IFL .EQ. 3)  GO TO 312
      DI=DIZZ
      DJ=DJZZ
      DK=DKZZ
      INDEX=1
      GO TO 104
C
  101 EFZZ=EF
      DI=DIZY
      DJ=DJZY
      DK=DKZY
      INDEX=2
      GO TO 104
  312 EFZZ=0.0
      EF=0.0
C
  102 EFZY=EF
      DI=DIYZ
      DJ=DJYZ
      DK=DKYZ
      INDEX=3
      IF ( IFL .EQ. 1)  GO TO 313
      GO TO 104
  313 EF=0.0
C
  103 EFYZ=EF
      DI=DIYY
      DJ=DJYY
      DK=DKYY
      INDEX=4
      IF ( IFL .EQ. 1)  GO TO 315
C
  104 USE6=  DEE*DEE +DCC
      USRA =  DKR/RCURV
      USK2 =  DK/DRR
      USHR =  DB/DRR
      USRR2=  USRR*USRR
      USDA =  DD/RCURV
      USP  =  USDA-1.0+ USHR2
      USER =  DEE/DRR
      USER2= USER*USER
      USER3= USER*USER2
      USER5= USER2*USER3
      FF  =  2.0* USKR* USER
     1     + 0.6666666*USER3*(DI*USRA -USHR*DJ + USKR*USP)
     2     + 0.4*USER5*( USKR*(1.0 - USRA*USRA*0.25 -2.0*USDA +USDA*USDA
     3                -3.0*USHR2*(1.0-USDA) +USHR2*USHR2) +
     4             DJ*(2.0*USHR*(1.0-USDA)- USHR*USHR2)   +
     5             DI*USRA*USP  )
C     WRITE (6,34)
C  34 FORMAT (1H0, 17HDI,DJ,DK,EF,INDEX)
C     WRITE (6,35) DJ,DK,EF,INDEX
C  35 FORMAT(1H0, 3F20.6 ,4X,I6)

      IF (INDEX  .EQ. 1)  GO TO 101
      IF (INDEX  .EQ. 2)  GO TO 102
      IF (INDEX  .EQ. 3)  GO TO 103
C
```

```
      EFYY=EF
      GO TO 110
  315 EFYY=0.0
      GO TO  110
  106 DNOM = DCC-DEE*DEE
      IF(DNOM  .GT. (-0.000001))  GO TO 107
      DP= (3.1415927  + ATAN(DEE*2.0*DRR/DNOM))/DRR
      GO TO 109
C
  107 IF(DNOM  .LT. (0.000001))   GO TO  108
      DP= (ATAN(DEE*2.0*DRR/DNOM))/DRR
      GO TO 109
C
  108 DP= 3.1415927/(2.0*DRR)
  109 IF(IFI .EQ. 3)  GO TO 212
C
      DI=DIZZ
      DJ=DJZZ
      DK=DKZZ
      INDEX=1
      GO TO 114
C
  111 EFZZ=EF
      DI=DIZY
      DJ=DJZY
      DK=DKZY
      INDEX=2
      GO TO 114
  212 EFZZ=0.0
      EF=0.0
C
  112 EFZY=EF
      DI=DIYZ
      DJ=DJYZ
      DK=DKYZ
      INDEX=3
      IF ( IFI .EQ. 1)  GO TO 213
      GO TO 114
  213 EF=0.0
C
  113 EFYZ=EF
      DI=DIYY
      DJ=DJYY
      DK=DKYY
      INDEX=4
      IF ( IFI .EQ. 1)  GO TO 215
C
C
  114 USE6=  DEE*DEE +DCC
      EF= DP*(-DCC*DI/RCURV -DJ*DB*0.5  +DK -0.75*DD*DJ*DB/RCURV +
     1    DD*DK/(2.0*RCURV))     +  2.0*DEE*DI/RCURV +(DEE*DB*DJ*(1.0
     2    +DD/RCURV)+DD*DJ*DB*DEE*(3.0*DEE*DEE+DCC)/(2.0*RCURV*USE6)
     3    -DD*DK*DEE/RCURV)/USE6
C
C      WRITE (6,34)

C     WRITE (6,35) DJ,DK,EF,INDEX
C
      IF(INDEX  .EQ. 1)  GO TO 111
      IF(INDEX  .EQ. 2)  GO TO 112
      IF(INDEX  .EQ. 3)  GO TO 113
      EFYY=EF
      GO TO 110
  215 EFYY=0.0
C     WRITE (6,30)
C  30 FORMAT (1H0,  13HCURVATURE , P)
C     WRITE(6,31) RCURV,  DP
C  31 FORMAT(1H0,2F20.6)
C     WRITE(6,32)
C  32 FORMAT(1H0,  19HEFZZ,EFZY,EFYZ,EFYY)
C     WRITE(6,33) EFZZ,EFZY,EFYZ,EFYY
C  33 FORMAT 1H0,4F20.6)
C
C
C     THE FORMATION OF F AND P BEGINS HERE.  REQUIRED ARE THE BESSEL
C     FUNCTIONS  J0,Y0,J1,Y1.   SIMPLE POLYNOMIALS WILL BE USED TO
C     EVALUATE THESE BESSEL FUNCTIONS.  TWO DIFFERENT EXPRESSIONS WILL
C     BE USED  DEPENDING ON  WHETHER  KBAR/3  IS GREATER OR LESS THAN 1
C     (KBAR= KR*M*R/C)  ACTUALLY  KBAR TIMES THE BESSEL FUNCTIONS ARE
C     CALCULATED.
C
  110 DK3 =DKK/3.0
      IF( DK3 .GT. 1.0) GO TO 200
      IF( DK3*DK3 .LT. 0.00000001) GO TO 150
      DLOG= ALOG(1.5*DK3)
      GO TO 160
  150 DLOG=0.0
C
C
  160 DK32= DK3*DK3
      DK34= DK32*DK32
      DK36= DK34*DK32
      DK38= DK34*DK34
      DK310= DK38*DK32
      DK312= DK310*DK32
C     WRITE(6,12)
C  12 FORMAT(1H0,  18HDK3,DK32,DK34,DK36)
C     WRITE(6,9) DK3,DK32,DK34,DK36
C     WRITE(6,13)
C  13 FORMAT(1H0, 16HDK38,DK310,DK312)
C     WRITE(6,14) DK38,DK310,DK312
C  14 FORMAT(1H0,3F20.6)
C     BESSEL  FUNCTIONS KJ1,KY1,KJ0,KY0 FOR SMALL ARGUMENT  K .LT. 3.0
      DKJ1=(DKK*DKK)*(0.5-0.56249985*DK32+0.21093573*DK34 -0.03954289 *
     1      DK36 +0.00443319*DK38 -0.00031761*DK310 +0.00001109*DK312)
      DKY1=(DKJ1*2.0*DLOG)/3.1415927     -0.6366198 + 0.2212091*DK32
     1      +2.1682709*DK34 -1.3164827*DK36 +0.3123951*DK38 -0.0400976
     2      *DK310 + 0.0027873*DK312
      DKJ0= DKK*( 1.0 - 2.2499997*DK32 +1.2656208*DK34 -0.3163866*DK36
     1    + 0.0444479*DK38 -0.0039444*DK310 +0.00021*DK312)
      DKY0= 2.0*DLOG*DKJ0/3.1415927 +DKK*(0.36746691+0.60559366*DK32
     1     -0.74350384*DK34 +0.25300117*DK36 -0.04261214*DK38 +
     2     0.00427916*DK310-0.00024846*DK312)
C     WRITE(6,15)
C  15 FORMAT(1H0,  36HTHE BESSEL FUNCTIONS KJ1,KY1,KJ0,KY0)

C     WRITE(6,9) DKJ1,DKY1,DKJ0,DKY0
      GO TO 300
C     BESSEL FUNCTIONS KJ1,KY1,KJ0,KY0 FOR LARGE ARG.   K .GT. 3.0
  200 DK31= 1.0/DK3
      DK312= DK31*DK31
      DK313= DK312*DK31
      DK314= DK313*DK31
      DK315= DK314*DK31
      DK316= DK315*DK31
C     WRITE(6,16)
C  16 FORMAT(1H0,41HDK312,DK313,DK314,DK316  WHERE DK31=1/DK3)
C     WRITE(6,9) DK312,DK313,DK314,DK316
      DF11=0.79788456 +0.00000156*DK31 +0.01659667*DK312 +0.00017105*
     1    DK313 -0.00249511*DK314 +0.00113653*DK315-0.00020033*DK316
      DTH1= DKK -2.35619449 +0.12499612*DK31 +0.00005650*DK312 -
     1      0.00637879*DK313 +0.00074348*DK314 +0.00079824*DK315 -
     2      0.00029166*DK316
      DF0 =0.79788456 -0.00000077*DK31  -0.00552740*DK312 -0.00009512*
     1     DK313 +0.00137237*DK314-0.00072805*DK315 +0.00014476*DK316
      DTH0= DKK -0.78539816 -0.04166397*DK31-0.00003954*DK312 +
     1     0.00262573*DK313 -0.00054125*DK314 -0.00029333*DK315
     2     +0.00013558*DK316
C     WRITE(6,17)
C  17 FORMAT(1H0,20HF1,THETA1, F0,THETA0)
C     WRITE(6,9)DF11,DTH1,DF0,DTH0
      SQDKK =  SQRT(DKK)
      DKJ1 = SQDKK*DF11*COS(DTH1)
      DKY1 = SQDKK*DF11*SIN(DTH1)
      DKJ0 = SQDKK*DF0*COS(DTH0)
      DKY0 = SQDKK*DF0*SIN(DTH0)
C     WRITE(6,18)
C  18 FORMAT(1H0, 53HWRITE OUT FAR FIELD BESSEL FUNCT. DKJ1,DKY1,DKJ0,DK
C    1Y0)
C     WRITE(6,9) DKJ1,DKY1,DKJ0,DKY0
C     THE QUANTITIES  RE F  IN F  RE P  AND  IN P   ARE NOW FORMED
  300  FMUR=-0.25*DKY1
      FMUI=-0.25*DKJ1
```

```
      PMUR= DM*0.25*DKJ0
      PMUI= DM*0.25*DKY0
C     WRITE(6,19)
C  19 FORMAT(1H0, 26HREAL F, IM.F ,REAL P, IM.P)
C     WRITE(6,9) FMUR,FMUI,PMUR,PMUI
C     THE SUMMATION MAY NOW BE PERFORMED FOR  FZZ,FZY,FYZ,FYY,MZZ,MZY,
C                                                           MYZ,MYY
      DRFZZ= DRFZZ +FMUR*EFZZ
      DIFZZ= DIFZZ +FMUI*EFZZ
      DRFZY= DRFZY +FMUR*EFZY
      DIFZY= DIFZY +FMUI*EFZY
      DRFYZ= DRFYZ +FMUR*EFYZ
      DIFYZ= DIFYZ +FMUI*EFYZ
      DRFYY= DRFYY +FMUR*EFYY
      DIFYY= DIFYY +FMUI*EFYY
      DRMZZ= DRMZZ +PMUR*EF1*EIM2
      DIMZZ= DIMZZ +PMUI*EF1*EIM2
      DRMZY= DRMZY +PMUR*EF1*EIM1
      DIMZY= DIMZY +PMUI*EF1*EIM1
      DRMYZ= DRMYZ -PMUR*EF2*EIM2
      DIMYZ= DIMYZ -PMUI*EF2*EIM2
      DRMYY= DRMYY -PMUR*EF2*EIM1

      DIMYY= DIMYY -PMUI*EF2*EIM1
C     WRITE(6,20)
C  20 FORMAT(1H0, 35HRUNNING SUM OF  RFZZ,IFZZ,RMZZ,IMZZ)
C     WRITE(6,9) DRFZZ,DIFZZ,DRMZZ,DIMZZ
C     WRITE(6,21)
C  21 FORMAT(1H0, 35H RUNNING SUM OF RFZY,IFZY,RMZY,IMZY)
C     WRITE(6,9) DRFZY,DIFZY,DRMZY,DIMZY
C     WRITE(6,22)
C  22 FORMAT(1H0,34HRUNNING SUM OF RFYZ,IFYZ,RMYZ,IMYZ)
C     WRITE(6,9) DRFYZ,DIFYZ,DRMYZ,DIMYZ
C     WRITE(6,23)
C  23 FORMAT(1H0, 34HRUNNING SUM OF RFYY,IFYY,RMYY,IMYY)
C     WRITE(6,9) DRFYY,DIFYY,DRMYY,DIMYY
 1001 CONTINUE
 1000 CONTINUE
      RETURN
      END


CORGAN          THIS ROUTINE IS USED TO REORGANIZE THE BOX AND BODY
C               ELEMENTS SO THAT THEIR LOADS MAY BE TRANSLATED TO THE
C               CORRECT SLENDER BODY ELEMENT.
C
      SUBROUTINE    ORGAN   ( ISTART,   ISTOP ,   NLBE  ,   NTP   ,
     *                X     ,   DELX  ,   PERCNT,   XLE   ,   XTE   ,
     *     XIS1,    XIS2,    ITYPE,    IPRNT  )
C
C ***    *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
C
C        ISTART          OUTPUT ARRAY CONTAINING THE FIRST S. B. ELEMENT
C                        THAT THE SENDING BOX OR ELEMENT CONTRIBUTES TO
C        ISTOP           OUTPUT ARRAY CONTAINING THE LAST S.B. ELE.
C        NLBE            NO. OF BODY ELEMENTS IN THE SLENDER BODY
C        NTP             NO. OF BOXES OR ELEMENTS SENDING LOADS
C
C        WHEN ITYPE = 0
C
C              X         ARRAY OF SENDING BOX OR ELEMENT LOCATION
C              DELX      ARRAY OF SENDING BOX OR ELEMENT LENGTH
C              PERCNT    PERCENT OF BOX LENGTH THAT  X  IS
C
C        **** ITYPE = 1
C
C              X         ARRAY OF SENDING S.B. ELEMENTS L.E.
C              DELX      ARRAY OF SENDING S.B. ELEMENTS T.E.
C              PERCNT    NOT USED
C
C        XLE             RESERVED
C        XTE             RESERVED
C        XIS1            L.E. OF RECIEVING BODY ELEMENTS
C        XIS2            T.E. OF RECIEVING BODY ELEMENTS
C        ITYPE           FLAG  = 0  SENDING ELEMENTS BOXES OR INT. ELEM.
C                              = 1  SENDING ELEMENTS ARE SLENDER BODY
C
C ***    *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *
C
      DIMENSION    ISTART (1)    , ISTOP  (1)    , X      (1)    ,
     *                DELX   (1)    , XIS1   (1)    , XIS2   (1)
C
      DATA     NPOT  /  6  /
C
C
      IF ( IPRNT  .EQ.  0  )  GO TO 5
      WRITE ( NPOT , 8000 ) ( I,  XIS1(I), XIS2(I), I = 1, NLBE  )
      WRITE ( NPOT , 8010 )
C
    5 CONTINUE
      IBE      =    1
      NLBE1    =    NLBE  +  1
C
      IF   (  NTP   .LE.  0 )  RETURN
      DO    2100  I = 1,  NTP
C
      ISTART (I)    =       0
      ISTOP  (I)    =       NLBE1
C
      IF ( ITYPE .EQ. 1 )    GO  TO  10
C                     CALCULATE  L.E. AND T.E. OF BOX OR I.B. ELEMENT
      XLEBOX           =     X(I) - PERCNT * DELX(I)
      XTEBOX           =     XLEBOX  +  DELX(I)
      GO  TO   20
C                     CALCULATE  L.E. AND T.E. OF SLENDER BODY ELEMENT
   10 XLEBOX           =     X(I)
      XTEBOX           =     DELX(I)
C                     CHECK FOR POSITION OF ELEMENT WITH RESPECT TO BODY
   20 IF (  XTEBOX  .LT.  XIS1( 1  )  )  GO TO 200
      IF (  XLEBOX  .GE.  XIS2(NLBE)  )  GO TO 200
C
      IF (  XLEBOX  .LT.  XIS1( 1  )  )  GO TO 250
C
C              LOCATE ELEMENT NUMBER CORR. TO L.E. OF BOX
   30 IF (  XLEBOX  .LT. XIS1(IBE))GO TO  300
      IF (  XLEBOX  .LT. XIS2(IBE))GO TO   40
      IBE             =     IBE  +  1
      IF   (  IBE  .GT. NLBE )  GO TO 200
      GO  TO   30
C
C                     CORRESPONDING ELEMENT FOUND  ( L.E. )
   40 ISTART ( I )    =     IBE
      GO  TO   1000
C                     BOX NOT INCLUDED IN BODY
  200 ISTOP (I)       =     0
      GO  TO   2000
C                     L.E. OF BOX OUT IN BODY
  250 IBE             =     1
      GO  TO   1000
C                     DECREMENT SB ELEMENT NO.
  300 IBE             =     IBE  -  1
      IF (  IBE  )  250,  250,  30
C
C          LOCATE ELEMENT NO. CORR. TO  T. E. OF BOX
C
 1000 IF   (   XTEBOX   .GT.   XIS2(IBE)  )     GO   TO   1200
      IF   (   XTEBOX   .GT.   XIS1(IBE)  )     GO   TO   1300
C
      IBE             =     IBE  -  1

      IF (  IBE  )  200,  200,  1000
C
 1200 IBE             =     IBE  +  1
      IF (  IBE  .LE.  NLBE )  GO TO  1000
C              ISTOP ALREADY SET = NLBE1
C              RESET IBE SO AS NOT TO EXCEED ARRAY NEXT TIME
      IBE             =     IBE  -  1
      GO  TO   2000
C
 1300 ISTOP(I)        =     IBE
C     GO  TO   2000
 2000 CONTINUE
C
      IF  (  IPRNT  .EQ.  0 )  GO TO 2100
      WRITE  ( NPOT,  8030 )  I,   X(I),      DELX(I),  XLEBOX,
     *           XTEBOX,   ISTART(I),          ISTOP(I)
C
 2100 CONTINUE
      RETURN
```

```
C
 8000 FORMAT  ( 44H1SUBROUTINE --ORGAN--    SLENDER BODY LIMITS   /
     * 48H0ELEMENT      X - L.E. (XIS1)      X - T.E. (XIS2)      /
     *       ( I6, 6X,  F14.5, 6X, F14.5 ) )
 8010 FORMAT  (  26H0LIMITS FOR APPLIED FORCES  //
     * 85H BOX/BODY          X              DELX          X - L.E.       X - T.E.
     *  FIRST S.B.  LAST S.B.  /
     *  9H ELEMENT, 57X,  20HELEMENT      ELEMENT     )
 8030 FORMAT  ( I8, 4E14.5, I6, 5X, I6 )
      END


CSBLOAD         THIS ROUTINE CALCULATES THE DISTRIBUTION OF THE LOADS ON
C               THE SLENDER BODIES.
C
      SUBROUTINE     SBLOAD  ( COEF     , IFIRST   , ILAST    , KS      ,
     *      XIS1     , XIS2     , DELTA    , X        , PERCNT   , XLE     ,
     *      XTE      , NBE      , FWZ      , FWY      , MWZ      , MWY     ,
     *      FZ       , FY       , MZ       , MY       , IPRNT              )
C
C ***      *     *     *     *     *     *     *     *     *     *     *     *     *
C
C          COEF       COEFFICIENT FOR FWZ, FWY,.. TO CALC. FZ, FY, ...
C          IFIRST     FIRST S.B.E. LOADS APPLIED TO
C          ILAST      LAST  S.B.E. LOADS APPLIED TO
C          XIS1       L.E. OF SLENDER BODY ELEMENTS
C          XIS2       T.E. OF SLENDER BODY ELEMENTS
C          DELTA      LENGTH OF SENDING ELEMENT OR BOX
C          X          COORDINATE OF SENDING ELEM. OR BOX
C          PERCNT     LOCATION OF X IN ELEM.
C          XLE        L.E. OF S.B.
C          XTE        T.E. OF S.B.
C          NBE        NO. SLENDER BODY ELEMENTS
C          FWZ        .
C          FWY        ..  LOADS ON SENDING ELEMENT
C          MWZ        ..     OR BOX.   MUST BE MULTIPLIED BY - COEF -
C          MWY        .
C          FZ         .

C          FY         ..  OUTPUT LOADS ON SLENDER BODY ELEMENTS
C          MZ         ..
C          MY         .
C
C ***      *     *     *     *     *     *     *     *     *     *     *     *     *
C
      COMPLEX        FWZ      , FWY      , FZB      , FYB      , MWZ      ,
     *    MWY      ,MZB       , MYB      , DELFZ    , DELFY    , FY       ,
     *    FZ,      MY,        MZ,        COEF
C
      DIMENSION      XIS1    (  1 ), XIS2    (  1 ), FZ      (  1 ),
     *               FY      (  1 ), MZ      (  1 ), MY      (  1 )
      DATA     NPOT   /  6  /
      IF  (  IPRNT   .NE.  0 )
     *  WRITE ( NPOT, 8000) COEF, FWZ, FWY,  MWZ,    MWY
      XLEBOX         =        X - PERCNT * DELTA
      XTEBOX         =        XLEBOX  +  DELTA
C               SET LIMITS SO THEY WILL NOT EXCEED LIMITS OF SLENDER BODY
      N1             =        MAX0 ( 1, IFIRST )
C
      N2             =        MIN0 ( NBE, ILAST )
C
  450 FZB      =      COEF  *    FWZ
      FYB      =      COEF  *    FWY
      MZB      =      COEF  *    MWZ
      MYB      =      COEF  *    MWY
C
      IF  (  IPRNT    .NE.  0)
     * WRITE ( NPOT,  490 )  FZB,  FYB,   MZB,    MYB
  480 FORMAT (43H0 LOADS ON SENDING ELEMENT, FZ, FY, MZ, MY /(4E14.5) )
C
C               DISTRIBUTE LOADS TO ELEMENTS OF SLENDER BODY
C
      DO      500     I  =  N1,  N2
C
C
C                    CALCULATE LIMITS OF CONTRIBUTION OF SENDING ELEMENT
      X1             =       XLEBOX
      X2             =       XTEBOX
      IF ( X1   .LT.XIS1(I)  )  X1   =  XIS1 (I)
      IF ( X2   .GT. XIS2(I) )  X2   =  XIS2(I)
C               FACTOR - PORTION OF SENDING ELE. APPLIED TO THIS S.B.E.
      FACTOR         =       (X2 - X1 ) / DELTA
      DELFZ          =    FACTOR  *    FZB
      DELFY          =    FACTOR  *    FYB
C               DIFERENCE OF CENTER OF PRESSURE LOCATION
      DX        =     ( ( XIS2(I) + XIS1(I) ) - ( X2 + X1 ) )  /  2.0
      FY(I)          =    FY(I )  +  DELFY
      FZ(I)          =    FZ(I )  +  DELFZ
      MZ(I)          =    MZ(I )  +  FACTOR *  MZB    +  DX * DELFZ
      MY(I)          =    MY(I )  +  FACTOR *  MYB    +  DX * DELFY
C
      IF ( IPRNT  .NE. 0 )
     *  WRITE ( NPOT, 490)  I,  DELFZ,    DELFY,   DX
  490 FORMAT ( I10, 5E14.5 )
C
  500 CONTINUE
C
      RETURN

C
C
 8000 FORMAT ( 6F12.4)
      END
```

```
      SUBROUTINE AERO(NMODE,NSTRIP,NW,  NBFM,NEWBFM,IBFS)
C   ***    COMPUTES  AERODYNAMIC PARAMETERS FOR ALL  LIFTING SURFACES
C          AND ALL SLENDER BODIES                                    S
      COMMON       NP,NTP,NB,N3Z,NBY,NTZ,NTY,FMACH,REFA,REFS,REFC,ND,NE,
     1             NDFA(10,2),NSBEA(10),NCARAY(100),NSARAY(100),NBARAY(100)
     2             ,ZB(10),YB(10),XLE(10),XTE(10),ARB(10),AVR(10),AC(200),
     3             AOP(200),XIS1(200),XIS2(200),XIJ(100),CS(210),YS(210),
     4             ZS(210),EE(200),SG(200),CG(200),X(500),XIC(500),KR,XM,
     5             DELX(500),XLAM(500),RIA(100),TH1A(100),TH2A(100),NFL(10)
     6             ,IFLA(30,2),NT12(10,2),NAS(100),NASB(200),YIN(100),
     7             ZIN(100),DETA(50),DZET(5)),NOBODY(50),NPANEL(50),NTO
     8             ,NTZS,NTYS
      COMPLEX      DCP(500),FZ( 200),FY( 200),MZ( 200),MY( 200), CN(200),
     1             CM(200),CZB(10),CYB(10),CMB(10),CNB(10),CZT,CYT,CMT,
     2             CNT, CXT, CLT, FZLB, MZLB, FYLB, MYLB
      COMPLEX      DFZX, DFYX, EIKMF, GBAR, PARTI
      DIMENSION  CPR(200),CPI(200)
      REAL         KR
   10 FORMAT  (1H1,50X, 7HMODE NO,I4// 2X, 9HSTRIP NC.,5X,1HY,12X,1HZ,
     1      9X,3HYOS,12X,16HLIFT COEFFICIENT,12X,18HMOMENT COEFFICIENT,
     2     6X,13HCP-R AND CP-I/)
   20 FORMAT  (1H0, 10HBODY ELEM.,5X,1HY,12X,1HZ, 9X, 3HXOL, 12X,
     1      16HLIFT COEFFICIENT,12X,18HMOMENT COEFFICIENT/)
   30 FORMAT  (1H0, 10HBODY ELEM.,5X,1HY,12X,1HZ, 9X, 3HXOL, 12X,
     1      50HMODIFIED LIFT COEFFICIENTS,  Z-  AND  Y-DIRECTIONS/)
   40 FORMAT  (1H0,    10H  BODY NO.,5X,1HY,12X,1HZ, 6X,8HX-CENTER, 9X,
     1      17HTOTAL LIFT COEFF.,11X, 19HTOTAL MOMENT COEFF./)
   50 FORMAT  (1H0, 4X,I4,3F12.4,4F14.6,2F10.5)
   60 FORMAT  (1H0,3X//14X,4HCZ =,2F16.6,10X,4HCY =,2F16.6//14X,4HCM =,
     1      2F16.6,10X,4HCN =,2F16.6//14X,4HCSL=,2F16.6//)
   70 FORMAT  (1H0,10X, 20I5/)
   80 FORMAT  (1H0,6F20.8)
      NTOT = NTP+NTZ+NTY
      NSB   = NB
      BETA2  = 1.0 - FMACH**2
      CONST  = 2.0 * KR * FMACH / BETA2
      SYMA = 1.0 + FLOAT(ND)
      SYMB = 1.0 - FLOAT(ND)
      DO   90  I=1,200
      CPR(I) = 0.0
      CPI(I) = 0.0
      FZ(I)= (0.0,0.0)
      FY(I)= (0.0,0.0)
      MZ(I)= (0.0,0.0)
      MY(I)= (0.0,0.0)
   90 CONTINUE
      NSBETO = 0
      DO  100  I=1,NB
  100 NSBETO = NSBETO + NSBEA(I)
      REWIND  NW
      IF  (NTZS.EQ.0.AND.NTYS.EQ.0)  GO TO  140
      REWIND  NEWBFM
      IF  (IBFS . NE . 0)  GO TO  140
      DO  130  JJ=1,NMODE
      REWIND  NBFM
      DO  120  IB=1,NB
      NSBE    = NSBEA(IB)
      IF  (NSBE. EQ .0)  GO TO  120
      DO  110  J =1,NMODE
      READ  (NBFM)  JMODE,NES,   (FZ(K), FY(K), MZ(K), MY(K), K=1,NSBE)
      IF (J.EQ.JJ) WRITE(NEWBFM) (FZ(K), FY(K), MZ(K), MY(K), K=1,NSBE)
  110 CONTINUE
  120 CONTINUE
  130 CONTINUE
      REWIND  NEWBFM
  140 CONTINUE
      DO  380  K=1,NMODE
      READ  (NW)   (DCP(I),  I=1,NTOT)
      I2    = 0
      LP    = 1
      WRITE  (6,10)  K
      DO  190  J=1,NSTRIP
      I1    = I2+1
      I2    = I2+NCARAY(LP)
      IF (I2.EQ.NBARAY(LP))  LP=LP+1
      XI14S = 0.25*CS(J) + XIJ(J)
      CN(J)= (0.0,0.0)
      CM(J)= (0.0,0.0)
      DO  150  I=I1,I2
      CN(J)= CN(J) + DCP(I)*DELX(I)
      CM(J)= CM(J) + DCP(I)*DELX(I)*(XIC(I) - XI14S)
  150 CONTINUE
      YOS   = YS(J)/REFS
      CH2     = CS(J)**2
      CN(J)   = CN(J)/CS(J)
      CM(J)   =-CM(J)/CH2
      ABSCNJ = SQRT((REAL(CN(J)))**2 + (AIMAG(CN(J)))**2)
      IF    ( ABSCNJ   .LE.0.000001)  GO TO  160
      CPR(J) =-REAL(CM(J))/REAL(CN(J)) + 0.25
      IF (KR.LE.0.0001)  GO TO  170
      CPI(J) =-AIMAG(CM(J))/AIMAG(CN(J))+0.25
      GO TO  180
  160 CONTINUE
      CPR(J) = 0.0
  170 CONTINUE
      CPI(J) = 0.0
  180 CONTINUE
      WRITE  (6,50)  J,YS(J),ZS(J),YOS,CN(J),CM(J),CPR(J),CPI(J)
  190 CONTINUE
      IF (NSB.EQ.0)  GO TO 290
      IF   (IBFS . NE . 0)  READ (NEWBFM) (FZ(LL),FY(LL),  LL=1,NSBETO)
      L2    = 0
      DO   280  N=1,NSB
      CZB(N) = (0.0,0.0)
      CYB(N) = (0.0,0.0)
      CMB(N) = (0.0,0.0)
      CNB(N) = (0.0,0.0)
      NSBE    = NSBEA(N)
      L1    = L2+1
      L2    = L2 + NSBEA(N)
      IF   (IBFS . NE . 0)  GO TO  200
      READ  (NEWBFM)   (FZ(LL),FY(LL),MZ(LL),MY(LL),  LL=L1,L2)
  200 CONTINUE
      SBL   = XIS2(L2) - XIS1(L1)
      WRITE  (6,20)
      DO  210  LB=L1,L2
      LX       = LX+1
      XSB   = 0.5*(XIS1(LB) + XIS2(LB))
      DXSB  = XIS2(LB) - XIS1(LB)
      XOL      = (XSB -XIS1(L1))/ SBL
      FZLB = FZ(LB)/ DXSB
      FYLB = FY(LB)/ DXSB
      MZLB = MZ(LB)/ DXSB
      MYLB = MY(LB)/ DXSB
      WRITE  (6,50)  LB, YB(N), ZB(N), XOL, FZLB, MZLB
      WRITE  (6,50)  LB, YB(N), ZB(N), XOL, FYLB, MYLB
      CZB(N) = CZB(N) + FZ(LB)
      CYB(N) = CYB(N) + FY(LB)
      CMB(N) = CMB(N) + (-FZ(LB)*(XSB-XIS1(L1)) + MZ(LB))
      CNB(N) = CNB(N) + (-FY(LB)*(XSB-XIS1(L1)) + MY(LB))
  210 CONTINUE
      LZYFL   = NDFA(N,2)
      AR       = ARB(N)
      WRITE  (6,30)
      DO   270  LB=L1,L2
      A2       = AVR(N)**2
      XSBT     = 0.5 * (XIS1(LB) + XIS2(LB))
      XOL      = (XSBT-XIS1(L1))/ SBL
      DFZX     = (0.0,0.0)
      DFYX     = (0.0,0.0)
      DO   260  KB=L1,L2
      XSBTB    = 0.5 * (XIS1(KB) + XIS2(KB))
      TRM1     = (XSBT - XSBTB)**2
      TRM2     = 0.5 * BETA2 * A2 * (1.0+AR)
      RZ       = SQRT(TRM1 + TRM2*AR)
      RY       = SQRT(TRM1 + TRM2)
      GO TO  (220,220,230), LZYFL
  220 CONTINUE
      RR       = RZ
      INDY     = 0
      GO TO  240
  230 CONTINUE
      RR       = RY
  240 CONTINUE
      FEXP     = ((XSBT - XSBTB)*FMACH  - RR) / REFC
      ARG      = CONST * FEXP
      EIKMF    = CMPLX(COS(ARG),    SIN(ARG))
      PARTR    = A2 * BETA2 * (1.0+AR) / (4.0 * RR*RR)
      PARTI    = (CMPLX( (1.0/RR), (CONST/REFC))) * EIKMF
```

```
      GBAR   = PARTR * PARTI                                         AERO1600
      IF  (LZYFL.EQ.3. OR . INDY.NE.0)  GO TO  250                  AERO1610
      DFZX   = DFZX + GBAR * FZ(KB)                                  AERO1620
      IF  (LZYFL.EQ.1)  GO TO  260                                   AERO1630
      INDY   = 1                                                     AERO1640
      GO TO  230                                                     AERO1650
  250 CONTINUE                                                       AERO1660
      DFYX   = DFYX + GBAR * FY(KB)                                  AERO1670
  260 CONTINUE                                                       AERO1680
      WRITE  (6,50)  LB, YB(N), ZB(N), XOL, DFZX, DFYX               AERO1690
  270 CONTINUE                                                       AERO1700
      CZB(N) = CZB(N)/REFA                                           AERO1710
      CYB(N) = CYB(N)/REFA                                           AERO1720
      CMB(N) = CMB(N)/(REFA*REFC)                                    AERO1730
      CNB(N) = CNB(N)/(REFA*REFC)                                    AERO1740
      XCEN   = (XIS1(L1) + XIS2(L2))/2.0                             AERO1750
      WRITE  (6,40)                                                  AERO1760
      WRITE  (6,50)   N, YB(N), ZB(N), XCEN, CZB(N), CMB(N)          AERO1770
      WRITE  (6,50)   N, YB(N), ZB(N), XCEN, CYB(N), CNB(N)          AERO1780
  280 CONTINUE                                                       AERO1790
  290 CONTINUE                                                       AERO1800
      CZT   = (0.0,0.0)                                              AERO1810
      CYT   = (0.0,0.0)                                              AERO1820
      CMT   = (0.0,0.0)                                              AERO1830
      CNT   = (0.0,0.0)                                              AERO1840
      CLT   = (0.0,0.0)                                              AERO1850
      DO  310  J=1,NSTRIP                                            AERO1860
      CH2   = CS(J)**2                                               AERO1870
      XI14S= 0.25*CS(J) + XI1(J)                                     AERO1880
      CMULT= 2.0*FE(J)*CG(J)                                         AERO1890
      SMULT= 2.0*FE(J)*SG(J)                                         AERO1900
      GUCJ = 1.0                                                     AERO1910
      IF  (ND.EQ.0)  GO TO  300                                      AERO1920
      IF  (ABS(YS(J)).LE.0.0001 . AND .ABS(CG (J)).LE.0.0001)  GUCJ=0.5 AERO1930
  300 CONTINUE                                                       AERO1940
      CXT   = CH2*CM(J) - CS(J)*CN(J)*(XI14S-XM)                     AERO1950
      CZT   = CZT + CS(J)*CN(J)*CMULT * GUCJ                         AERO1960
      CYT   = CYT + CS(J)*CN(J)*SMULT * GUCJ                         AERO1970
      CMT   = CMT + CXT * CMULT        * GUCJ                        AERO1980
      CNT   = CNT + CXT * SMULT        * GUCJ                        AERO1990
      CLT   = CLT+CS(J)*CN(J)*(YS(J)*CG (J)+ZS(J)*SG (J))*2.0*FE(J)*GUCJ AERO2000
  310 CONTINUE                                                       AERO2010
      CZT   = CZT/REFA                                               AERO2020
      CYT   = CYT/REFA                                               AERO2030
      CMT   = CMT/(REFA*REFC)                                        AERO2040
      CNT   = CNT/(REFA*REFC)                                        AERO2050
      IF (NTZS . EQ . 0)  GO TO  330                                 AERO2060
      L1    = 0                                                      AERO2070
      L2    = 0                                                      AERO2080
      J1    = 1                                                      AERO2090
      J2    = NBZ                                                    AERO2100
      DO  320  J=J1,J2                                               AERO2110
      GLCB = 1.0                                                     AERO2120
      IF  (ABS(YB( J)).LE.0.0001)  GLCB = 0.5                        AERO2130
      L1    = L2+1                                                   AERO2140
      L2    = L2+NSBFA(J)                                            AERO2150
      CZT   = CZT + CZB(J) * GLCB                                    AERO2160
      CMT   = CMT + (CMB(J)-CZB(J)*(XIS1(L1)-XM)/REFC) * GLCB        AERO2170
      CLT   = CLT + REFA * CZB(J) * YB(J) * GLCB                     AERO2180
  320 CONTINUE                                                       AERO2190
  330 CONTINUE                                                       AERO2200
      IF (NTYS . EQ . 0)  GO TO  370                                 AERO2210
      L1    = 0                                                      AERO2220
      L2    = 0                                                      AERO2230
      J1    = NB-NBY+1                                               AERO2240
      J2    = NB                                                     AERO2250
      J1M1 = J1-1                                                    AERO2260
      IF  (J1M1.EQ.0)  GO TO  350                                    AERO2270
      DO  340  JX=1,J1M1                                             AERO2280
      L2    = L2+NSBFA(JX)                                           AERO2290
  340 CONTINUE                                                       AERO2300
  350 CONTINUE                                                       AERO2310
      DO  360  J=J1,J2                                               AERO2320
      GLCB = 1.0                                                     AERO2330
      IF  (ABS(YB( J)).LE.0.0001)  GLCB = 0.5                        AERO2340
      L1    = L2+1                                                   AERO2350
      L2    = L2+NSBFA(J)                                            AERO2360
      CYT   = CYT + CYB(J) * GLCB                                    AERO2370
      CNT   = CNT + (CNB(J)-CYB(J)*(XIS1(L1)-XM)/REFC) * GLCB        AERO2380
      CLT   = CLT + REFA * CYB(J) * ZB(J) * GLCB                     AERO2390
  360 CONTINUE                                                       AERO2400
  370 CONTINUE                                                       AERO2410
      CZT   = SYMA * CZT                                             AERO2420
      CYT   = SYMB * CYT                                             AERO2430
      CMT   = SYMA * CMT                                             AERO2440
      CNT   = SYMB * CNT                                             AERO2450
      CLT   = SYMB * CLT / (2.0*REFS*REFA)                           AERO2460
      WRITE  (6,60)  CZT,CYT,CMT,CNT,CLT                             AERO2470
  380 CONTINUE                                                       AERO2480
      RETURN                                                         AERO2490
      END                                                            AERO2500
```

```
      SUBROUTINE GENF(NMODE,NSTRIP,NW,NEWDFM,IMODE,AA,NAI,NPR2,IBFS)    GENF0010
C   ***   COMPUTES GENERALIZED FORCES FOR ALL PRESSURE-  AND DEFLECTION GENF0020
C         MODES. TWO OPTIONS AVAILABLE  --  AGARD GENERALIZED FORCES    GENF0030
C         ARE COMPUTED IF  N2=1  (INPUT), CONVENTIONAL FORCES IF N2=2   GENF0040
      COMMON      NP,NTP,NB,NBZ,NBY,NTZ,NTY,FMACH,REFA,REFS,REFC,ND,NE, GENF0050
     1            NBFA(10,2),NSBFA(10),NCARAY(100),NSARAY(100),NBARAY(100)GENF0060
     2            ,ZB(10),YB(10),XLE(10),XTE(10),ARB(10),AVR(10),AC(200), GENF0070
     3            AOP(200),XIS1(200),XIS2(200),XIJ(100),CS(210),YS(210), GENF0080
     4            ZS(210),EE(200),SG(200),CG(200),X(500),XIC(500),KR,XM, GENF0090
     5            DELX(500),XLAM(500),RIA(100),THIA(100),TH2A(100),NFL(10)GENF0100
     6            ,IFLA(30,2),NT12(10,2),NAS(100),NASB(200),YIN(100),   GENF0110
     7            ZIN(100),DETA(50),DZET(50),NOBODY(50),NPANEL(50),NTO  GENF0120
     8            ,NTZS,NTYS                                            GENF0130
      DIMENSION  HQA(700),DHQ(200),IMODE(2,150,3),  AA(2,150,3),NAI(3)  GENF0140
      COMPLEX     DCP(500),FZ( 200),FY( 200),MZ( 200),MY( 200),         GENF0150
     1            QW(50),QZ(50),QY(50),QIJ(50)                          GENF0160
   10 FORMAT  (1H1,45X, 7HMODE NO,I4//3X,8HPANEL NO, 10H  STRIP NO,     GENF0170
     1         10H    BOX NO,8X,3HXOC,9X,1HX,11X,1HY,11X,1HZ,15X,       GENF0180
     2         10H PRESSURES/ )                                         GENF0190
   20 FORMAT  (1H1,30X,3HTHE,I4,33H MODIFIED H-ELEMENTS FOR MODE NO.,I4)GENF0200
   30 FORMAT  (1H0,33X, 27HNO COEFFICIENT FOR MODE NO.,  I4/)           GENF0210
   40 FORMAT  (//30X,3HTHE,I4,38H MODIFIED  CH/DX-ELEMENTS FOR MODE NO.,GENF0220
     1         I4 )                                                     GENF0230
   50 FORMAT  (1H0,5X,I4,6X,I4,6X,I4,3X, 4F12.5,2F12.6 )                GENF0240
   60 FORMAT  (6I10)                                                    GENF0250
   70 FORMAT  (1H0,6F15.5)                                              GENF0260
   80 FORMAT  (1H1,5X,15HDEFLECTION MODE,5X,13HPRESSURE MODE,25X,       GENF0270
     1         19HGENERALIZED FORCES/56X,11HPANELS ONLY,14X, 5HTOTAL/)  GENF0280
   90 FORMAT  (1H0,7X,I4,19X,I4,5X,2F18.8,3X,2E18.8)                    GENF0290
      NBOX = NTP                                                        GENF0300
      IZ   = 0                                                          GENF0310
      CNOT = 1.0/(     REFA*REFC)                                       GENF0320
      IF  (NPR2.EQ.1)  CNOT= -1.0/(2.0*(REFS**3))                       GENF0330
      NTOT = NTP+NTZ+NTY                                                GENF0340
      NTOTS= NTP+NTZS+NTYS                                              GENF0350
      MARK = 0                                                          GENF0360
      DO  100  I=1,200                                                  GENF0370
      DHQ(I) = 0.0                                                      GENF0380
  100 CONTINUE                                                          GENF0390
      IF  (NB . EQ . 0)  GO TO  130                                     GENF0400
      NSBFTO = 0                                                        GENF0410
      DO  110  I=1,NB                                                   GENF0420
  110 NSBFTO = NSBFTO + NSBFA(I)                                        GENF0430
      DO  120  I=1,200                                                  GENF0440
      FZ(I)  = (0.0,0.0)                                                GENF0450
      FY(I)  = (0.0,0.0)                                                GENF0460
      MZ(I)  = (0.0,0.0)                                                GENF0470
      MY(I)  = (0.0,0.0)                                                GENF0480
  120 CONTINUE                                                          GENF0490
  130 CONTINUE                                                          GENF0500
      KQ   = 0                                                          GENF0510
      KQOLD= 0                                                          GENF0520
      DO 730  K=1,NMODE                                                 GENF0530
      KQ   = KQOLD                                                      GENF0540
      REWIND  NW                                                        GENF0550
      DO  140  KRUN=1,K                                                 GENF0560
      READ  (NW)  (DCP(IK),  IK=1,NTOT)                                 GENF0570
  140 CONTINUE                                                          GENF0580
      I2   = 0                                                          GENF0590
      LP   = 1                                                          GENF0600
      WRITE  (6,10)  K                                                  GENF0610
      DO 160  J=1,NSTRIP                                                GENF0620
      I1   = I2+1                                                       GENF0630
      I2   = I2+NCARAY(LP)                                              GENF0640
      DO 150  I=I1,I2                                                   GENF0650
      XOC  = (XIC(I)-XIJ(J)) / CS(J)                                    GENF0660
      WRITE  (6,50)  LP,J,I,XOC,XIC(I),YS(J),ZS(J), DCP(I)              GENF0670
  150 CONTINUE                                                          GENF0680
      IF (I2.EQ.NBARAY(LP))  LP=LP+1                                    GENF0690
  160 CONTINUE                                                          GENF0700
      IF  (NPR2.EQ.2) GO TO  730                                        GENF0710
      J2   = 0                                                          GENF0720
      I2   = 0                                                          GENF0730
      DO  170  I=1,NBOX                                                 GENF0740
      HQA(I) = 0.0                                                      GENF0750
  170 CONTINUE                                                          GENF0760
      IF  (NAI(1).EQ.0)  GO TO  370                                     GENF0770
      KQ   = KQ+1                                                       GENF0780
      MODE = IMODE(1,KQ,1)                                              GENF0790
      NQQ  = MODE/1000000                                               GENF0800
      NQMK = NQQ - K                                                    GENF0810
      IF  (NQMK)180,190,370                                             GENF0820
  180 CONTINUE                                                          GENF0830
      WRITE  (6,30)  K                                                  GENF0840
      GO TO  370                                                        GENF0850
  190 CONTINUE                                                          GENF0860
      DO  360  L=1,NP                                                   GENF0870
      KQT  = 0                                                          GENF0880
  200 CONTINUE                                                          GENF0890
      MODE = IMODE(1,KQ,1)                                              GENF0900
      NQQ  = MODE/1000000                                               GENF0910
      NQMK = NQQ - K                                                    GENF0920
      IF  (NQMK)360,210,240                                             GENF0930
  210 CONTINUE                                                          GENF0940
      NPP  = MODE/1000 - (MODE/1000000)*1000                            GENF0950
      NPML = NPP - L                                                    GENF0960
      IF (NPML)230,220,245                                              GENF0970
  220 CONTINUE                                                          GENF0980
      KQT  = KQT+1                                                      GENF0990
  230 CONTINUE                                                          GENF1000
      KQ   = KQ+1                                                       GENF1010
      IF  (KQ.LE.(NAI(1)))   GO TO  200                                 GENF1020
  240 CONTINUE                                                          GENF1030
      KQ   = KQ-1                                                       GENF1040
  245 NA2  = KQ                                                         GENF1050
      J1   = J2+1                                                       GENF1060
      J2   = J2+NSARAY(L)                                               GENF1070
      NA1  = NA2 - KQT + 1                                              GENF1080
      IF  (KQT.NE.0)  GO TO  250                                        GENF1090
      I2   = NBARAY(L)                                                  GENF1100
      GO TO  360                                                        GENF1110
  250 CONTINUE                                                          GENF1120
      DO  350  J=J1,J2                                                  GENF1130
      GUCJ = 2.0                                                        GENF1140
      IF  (ABS(YS(J)).LE.0.0001 . AND .ABS(CG (J)).LE.0.0001)  GUCJ=1.0 GENF1150
      I1   = I2+1                                                       GENF1160
      I2   = I2+NCARAY(L)                                               GENF1170
      DO  340  I=I1,I2                                                  GENF1180
      HQA(I) = 0.0                                                      GENF1190
      DELA   = 2.0*DELX(I)*EE(J)                                        GENF1200
      X4OC   = XIC(I)/REFC                                              GENF1210
      DO  330  NA=NA1,NA2                                               GENF1220
      MODE = IMODE(1,NA,1)                                              GENF1230
      NB   = MODE - (MODE/10)*10                                        GENF1240
      N    = MODE/10 - (MODE/100)*10                                    GENF1250
      M    = MODE/100 - (MODE/1000)* 10                                 GENF1260
      AMODE= AA(2,NA,1)                                                 GENF1270
      YINNB = 0.0                                                       GENF1280
      ZINNB = 0.0                                                       GENF1290
      IF  (NB.EQ.0)  GO TO 260                                          GENF1300
      YINNB = NB*YIN(L)                                                 GENF1310
      ZINNB = NB*ZIN(L)                                                 GENF1320
  260 CONTINUE                                                          GENF1330
      Y4OC = SQRT((YS(J)-   YINNB )**2 + (ZS(J)-   ZINNB )**2)          GENF1340
      DO  320  MI=1,6                                                   GENF1350
      MIM  = MI-1                                                       GENF1360
      DO  310  NI=1,6                                                   GENF1370
      NIM  = NI-1                                                       GENF1380
      AQMN = 0.0                                                        GEN 1390
      IF (M.EQ.MIM.AND.N.EQ.NIM)  AQMN=AMODE                            GENF1400
      IF  (ABS(AQMN).LE.0.0001)  GO TO  310                             GENF1410
      IF  (ABS(X4OC).LE.0.0001.AND.NIM.EQ.0)  GO TO  270                GENF1420
      X4TRM  = X4OC**NIM                                                GENF1430
      GO TO  280                                                        GENF1440
  270 CONTINUE                                                          GENF1450
      X4TRM  = 1.0                                                      GENF1460
  280 CONTINUE                                                          GENF1470
      IF  (ABS(Y4OC).LE.0.0001.AND.MIM.EQ.0)  GO TO  290                GENF1480
      Y4TRM  = Y4OC**MIM                                                GENF1490
      GO TO  300                                                        GENF1500
  290 CONTINUE                                                          GENF1510
      Y4TRM  = 1.0                                                      GENF1520
  300 CONTINUE                                                          GENF1530
      HQA(I) = HQA(I) + AQMN* X4TRM * Y4TRM                             GENF1540
  310 CONTINUE                                                          GENF1550
  320 CONTINUE                                                          GENF1560
  330 CONTINUE                                                          GENF1570
      HQA(I) = REFC*DELA*HQA(I) * GUCJ                                  GENF1580
  340 CONTINUE                                                          GENF1590
```

```
350 CONTINUE                                                    GENF1600
360 CONTINUE                                                    GENF1610
370 CONTINUE                                                    GENF1620
    KOOLD= KQ                                                   GENF1630
    IF (NB.EQ.0)  GO TO  590                                    GENF1640
    NBXP1= NBOX+1                                               GENF1650
    DO  380  I=NBXP1,NTOTS                                      GENF1660
    HOA(I) = 0.0                                                GENF1670
380 CONTINUE                                                    GENF1680
    DO  390  I=1,200                                            GENF1690
    DHQ(I) = 0.0                                                GENF1700
390 CONTINUE                                                    GENF1710
    IF  (NAI(2).EQ.0.AND.NAI(3).EQ.0)  GO TO  590               GENF1720
400 CONTINUE                                                    GENF1730
    MARK = 0                                                    GENF1740
    KO   = 0                                                    GENF1750
    I2   = NBOX                                                 GENF1760
    LB1  = 1                                                    GENF1770
    LB2  = NBZ                                                  GENF1780
    LBO  = 2                                                    GENF1790
    IF (NBZ.NE.0)  GO TO  420                                   GENF1800
    LB2  = NBY                                                  GENF1810
    MARK = 1                                                    GENF1820
410 CONTINUE                                                    GENF1830
    LBO  = 3                                                    GENF1840
420 CONTINUE                                                    GENF1850
    DO  580  LB=LB1,LB2                                         GENF1860
    IG   = 0                                                    GENF1870
    IF  (LB . EQ . 1)  GO TO  430                               GENF1880
    LBM1 = LB - 1                                               GENF1890
    IG   = NSBFA(LBM1)                                          GENF1900
430 CONTINUE                                                    GENF1910
    KOT  = 0                                                    GENF1920
    KO   = KO+1                                                 GENF1930
440 CONTINUE                                                    GENF1940
    MODE = IMODE(1,KO,LBO)                                      GENF1950
    NQO  = MODE/1000000                                         GENF1960
    NOMK = NQO-K                                                GENF1970
    IF  (NOMK)480,460,490                                       GENF1980
450 CONTINUE                                                    GENF1990
    IF (K.EQ.NMODE)  GO TO  490                                 GENF2000
    WRITE  (6,30)  K                                            GENF2010
    GO TO  590                                                  GENF2020
460 CONTINUE                                                    GENF2030
    NAB  = MODE/1000 - (MODE/1000000)*1000                      GENF2040
    NAML = NAB - LB                                             GENF2050
    IF (NAML)450,470,490                                        GENF2060
470 CONTINUE                                                    GENF2070
    KOT  = KOT+1                                                GENF2080
480 CONTINUE                                                    GENF2090
    KO   = KO+1                                                 GENF2100
    IF  (KO.LE.(NAI(LBO))) GO TO  440                           GENF2110
490 CONTINUE                                                    GENF2120
    KO   = KO -1                                                GENF2130
    NA2  = KO                                                   GENF2140
    I1   = I2+1                                                 GENF2150
    I2   = I2+NSBFA(LB)                                         GENF2160
    NA1  = NA2 - KOT + 1                                        GENF2170
    IF  (KOT.EQ.0)  GO TO  580                                  GENF2180
    DO  570  IB=I1,I2                                           GENF2190
    HOA(IB) = 0.0                                               GENF2200
    IZ   = IB-NTP                                               GENF2210
    I    = IG + 1                                               GENF2220
    DHQ(IZ) = 0.0                                               GENF2230
    XIAV = (XIS1(IG) + XIS2(IG)) / 2.0                          GENF2240
    XCLC = XIAV/REFC                                            GENF2250
    DO  560  NA=NA1,NA2                                         GENF2260
    MODE = IMODE(1,NA,LBO)                                      GENF2270
    NB   = 0                                                    GENF2280
    M    = 0                                                    GENF2290
    N    = MODE/100 - (MODE/1000)*10                            GENF2300
    BMODE= AA(2,NA,LBO)                                         GENF2310
    YCB  = 1.0                                                  GENF2320
    DO  540  NI=1,6                                             GENF2330
    NIM  = NI-1                                                 GENF2340
    ABQM = 0.0                                                  GENF2350
    IF                (N.EQ.NIM)  ABQM=BMODE                    GENF2360
    IF  (ABS(XCLC).LE.0.0001.AND.NIM.EQ.0)  GO TO  500          GENF2370
    XCTRM1 = XCLC**NIM                                          GENF2380
    GO TO  510                                                  GENF2390
500 CONTINUE                                                    GENF2400
    XCTRM1 = 1.0                                                GENF2410
510 CONTINUE                                                    GENF2420
    HOA(IB) = HOA(IB) + ABQM * XCTRM1                           GENF2430
    IF  (NIM.EQ.0)  GO TO  540                                  GENF2440
    NIM2 = NI-2                                                 GENF2450
    IF  (ABS(XCLC).LE.0.0001.AND.NIM2.EQ.0)  GO TO  520         GENF2460
    XCTRM2 = XCLC**NIM2                                         GENF2470
    GO TO  530                                                  GENF2480
520 CONTINUE                                                    GENF2490
    XCTRM2 = 1.0                                                GENF2500
530 CONTINUE                                                    GENF2510
    DHQ(IZ) = DHQ(IZ) + NIM*ABQM * XCTRM2                       GENF2520
540 CONTINUE                                                    GENF2530
550 CONTINUE                                                    GENF2540
560 CONTINUE                                                    GENF2550
    HOA(IB) = REFC*HOA(IB)                                      GENF2560
    IF (YB(LB).EQ.0)  GO TO  570                                GENF2570
    HOA(IB) = 2.0*HOA(IB)                                       GENF2580
    DHQ(IZ) = 2.0*DHQ(IZ)                                       GENF2590
570 CONTINUE                                                    GENF2600
580 CONTINUE                                                    GENF2610
    IF  (MARK.NE.0)  GO TO  590                                 GENF2620
    IF (NBY.EQ.0)   GO TO  590                                  GENF2630
    MARK = 1                                                    GENF2640
    LB1  = NB-NBY+1                                             GENF2650
    LB2  = NB                                                   GENF2660
    KO   = 0                                                    GENF2670
    GO TO  410                                                  GENF2680
590 CONTINUE                                                    GENF2690
    NSBETO = IZ                                                 GENF2700
    WRITE  (6,20)  NTOTS , K                                    GENF2710
    WRITE  (6,70)  (HOA(I),  I=1,NTOTS)                         GENF2720
    IF  (NSBETO.EQ.0)  GO TO  600                               GENF2730
    WRITE  (6,40)  NSBETO, K                                    GENF2740
    WRITE  (6,70)  (DHQ(I),  I=1,NSBETO)                        GENF2750
600 CONTINUE                                                    GENF2760
    DO  610  KX=1,50                                            GENF2770
    QW(KX) = (0.0,0.0)                                          GENF2780
    QZ(KX) = (0.0,0.0)                                          GENF2790
    QY(KX) = (0.0,0.0)                                          GENF2800
    QIJ(KX)= (0.0,0.0)                                          GENF2810
610 CONTINUE                                                    GENF2820
    REWIND  NW                                                  GENF2830
    REWIND  NEWBEM                                              GENF2840
    WRITE  (6,80)                                               GENF2850
    IZYO = 0                                                    GENF2860
    DO  720  KK=1,NMODE                                         GENF2870
    READ  (NW)  (DCP(IX),  IX=1,NTOT)                           GENF2880
    DO  620  II=1,NBOX                                          GENF2890
    QW(KK) = QW(KK) +      DCP(II)*HOA(II)                      GENF2900
620 CONTINUE                                                    GENF2910
    QW(KK)   = CNDT*QW(KK)                                      GENF2920
    IF  (NB.EQ.0)  GO TO  710                                   GENF2930
    IF  (IBES . NE . 0)  GO TO  640                             GENF2940
    L2      = 0                                                 GENF2950
    DO  630  NBX=1,NB                                           GENF2960
    NSBE    = NSBFA(NBX)                                        GENF2970
    IF  (NSBE. EQ . 0)  GO TO  630                              GENF2980
    L1      = L2 + 1                                            GENF2990
    L2      = L2 + NSBE                                         GENF3000
    READ  (NEWBEM)  (FZ(LL),FY(LL),MZ(LL),MY(LL),  LL=L1,L2)    GENF3010
630 CONTINUE                                                    GENF3020
    GO TO  650                                                  GENF3030
640 READ  (NEWBEM)  (FZ(LL), FY(LL),  LL=1,NSBETO)              GENF3040
650 CONTINUE                                                    GENF3050
    IB1  = NBOX+1                                               GENF3060
    IB2  = NBOX+NTZS                                            GENF3070
    IZY  = 0                                                    GENF3080
    IDH  = 0                                                    GENF3090
    QZ(KK) = (0.0,0.0)                                          GENF3100
    IF (NTZS.EQ.0)  GO TO  670                                  GENF3110
    DO  660  II=IB1,IB2                                         GENF3120
    IZY  = IZY+1                                                GENF3130
    IDH  = IDH+1                                                GENF3140
    QZ(KK) = QZ(KK) + FZ(IZY)*HOA(II) + MZ(IZY)*DHQ(IDH)        GENF3150
660 CONTINUE                                                    GENF3160
670 CONTINUE                                                    GENF3170
    IF (NTYS.EQ.0)  GO TO  710                                  GENF3180
    IB1  = IB2+1                                                GENF3190
```

```
      IB2   = IB2+NTYS
      IZY   = 0
      LAYE    = NA - NAY + 1
      IF (LAYE . EQ . 1)  GO TO  690
      LEM1    = LAYE - 1
      DO 680  LY = 1,LEM1
  680 IZY     = IZY + NSPFA(LY)
  690 CONTINUE
      QY(KK) = (0.0,0.0)
      DO  700  II=IB1,IB2
      IZY   = IZY+1
      IDH   = IDH+1
      QY(KK) = QY(KK) + FY(IZY)*HDA(II) + HY(IZY)*DHQ(IDH)
  700 CONTINUE
  710 CONTINUE
      QIJ(KK) = QN(KK) + CNOT*(QZ(KK) + QY(KK))
      WRITE  (6,90)  K,KK, QN(KK), QIJ(KK)
      IZYO = IZY
  720 CONTINUE
  730 CONTINUE
      RETURN
      END
```

References

1. Giesing, J. P., Kalman, T. P., and Rodden, W. P., "Subsonic Unsteady Aerodynamics for General Configurations: Part II, Volume I - Application of the Doublet-Lattice Method and the Method of Images to Lifting-Surface/Body Interference," AFFDL-TR-71-5, Part II, Volume I, August 1971, Air Force Flight Dynamics Laboratory, Wright-Patterson Air Force Base, Ohio.

2. Giesing, J. P., Kalman, T. P., and Rodden, W. P., "Subsonic Unsteady Aerodynamics for General Configurations: Part I, Volumes I and II - Direct Application of the Nonplanar Doublet-Lattice Method," AFFDL-TR-71-5, Part I, Volumes I and II, February 1971, Air Force Flight Dynamics Laboratory, Wright-Patterson Air Force Base, Ohio.

3. Hess, J. L., and Riddell, T. M., "Direct Solution of a Square Matrix Whose Size Greatly Exceeds High-Speed Storage," Douglas Aircraft Company Report No. DAC-70000, July 1969.

4. Abramowitz, M., Stegun, I. A., "Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables", National Bureau of Standards Applied Mathematics Series 55, March 1965.

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Douglas Aircraft Company Long Beach, California	2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED 2b GROUP N/A

3. REPORT TITLE

Subsonic Unsteady Aerodynamics for General Configurations
Part II, Volume II. Computer Program N5KA

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

December 1969 to August 1971

5. AUTHOR(S) (Last name, first name, initial)

J. P. Giesing, T. P. Kalman, W. P. Rodden

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
APRIL 1972	243	60

8a. CONTRACT OR GRANT NO. F33615-70-C-1167 b. PROJECT NO. 1370 c. Task No. 137003 d.	9a. ORIGINATOR'S REPORT NUMBER(S) AFFDL-TR-71-5, Part II, Volume II 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) Douglas Report No. MDC-J0944

10. AVAILABILITY/LIMITATION NOTICES

Approved for public release; distribution is unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	Air Force Flight Dynamics Laboratory Wright-Patterson AFB, Ohio 45433

13. ABSTRACT

A technique for predicting steady and oscillatory aerodynamic loads on general configurations has been developed which is based on the Doublet-Lattice Method and the method of images. Chord- and spanwise loading on lifting surfaces and longitudinal body load distributions are determined. Configurations may be composed of an assemblage of bo   lliptic cross sections and a distribution of width or radius) and lifting . aces (arbitrary planform and dihedral, with or without control surfaces). Loadings predicted by this method are required for flutter, gust, frequency response and static aeroelastic analyses and may be used to determine static and dynamic stability derivatives.

The methods described in this report are intended to be used by airplane designers to calculate with improved accuracy, the unsteady aerodynamic pressures that act on a lifting surface being propelled at subsonic speeds. The new feature of these calculations is that the effects on the pressure field induced by interference between the fuselage, for example, and the wing or the wing, pylon and nacelle, are taken into account. These calculations are an essential ingredient of flutter analyses and will improve the confidence level of such calculations in preventing wing-store flutter and flutter of advanced vehicles where fuselages are relatively large, provide some lifting capability and cause noticeable interference effects. The general requirement for such calculations are contained in Military Specification MIL- -8870A(USAF).

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
subsonic lifting surface theory						
wing-body interference						
aircraft flutter						
dynamic loads analysis						
nonplanar lifting surface theory						
vortex lattice						
doublet lattice						
fuselage and nacelle loads						
generalized forces						
control surface aerodynamic loads						
oscillatory aerodynamics						
unsteady aerodynamic loads						

INSTRUCTIONS

1. ORIGINATING ACTIVITY. Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2a. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2b. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S). Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE. Enter the date of the report as day, month, year; or month, year. If more than one date appears on the report, use date of publication.

7a. TOTAL NUMBER OF PAGES. The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7b. NUMBER OF REFERENCES. Enter the total number of references cited in the report.

8a. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8b, 8c, & 8d. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9a. ORIGINATOR'S REPORT NUMBER(S). Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9b. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through
________________________________."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through
________________________________."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through
________________________________."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES. Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS), (S), (C), or (U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, roles, and weights is optional.

☆U.S.Government Printing Office: 1972 — 759-085/545